



**United States Military Academy  
West Point, New York 10996**

**WARFIGHTER'S SIMULATION 2000  
TRAINING REQUIREMENTS ANALYSIS  
PROGRAM**

**OPERATIONS RESEARCH CENTER  
TECHNICAL REPORT**

**Major Robert G. Phelan, Jr.  
Captain Michael McNett  
Lieutenant Colonel Michael L. McGinnis**

**June 1997**

The Operations Research Center is supported by the Assistant Secretary of the Army for Financial Management & Comptroller. The Sponsor for this project is the U.S. Army Simulation, Training and Instrumentation Command (STRICOM), Orlando, Florida, and the National Simulation Center, Fort Leavenworth, Kansas.

**DTIC QUALITY INSPECTED 4**

**DISTRIBUTION STATEMENT A**  
Approved for Public Release  
Distribution Unlimited

19990325 027

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.				
1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE JUNE 1997	3. REPORT TYPE AND DATES COVERED TECHNICAL REPORT	
4. TITLE AND SUBTITLE  WARFIGHTER'S SIMULATION 2000 TRAINING REQUIREMENTS ANALYSIS PROGRAM			5. FUNDING NUMBERS	
6. AUTHOR(S) MAJ ROBERT G. PHELAN, JR. CPT MICHAEL McNETT LTC MICHAEL L. McGINNIS				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)  USMA OPERATIONS RESEARCH CENTER WEST POINT, NEW YORK 10996-1779			8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)			10. SPONSORING / MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES				
12a. DISTRIBUTION / AVAILABILITY STATEMENT  DISTRIBUTION STATEMENT A. APPROVED FOR PUBLIC RELEASE; DISTRIBUTION IS UNLIMITED.			12b. DISTRIBUTION CODE	
13. ABSTRACT (Maximum 200 words)  THIS REPORT DISCUSSES THE DESIGN AND IMPLEMENTATION OF A MULTI-USER DATABASE SYSTEM CALLED THE TRAINING REQUIREMENTS ANALYSIS PROGRAM (T-RAP), ON THE WORLD WIDE WEB THAT THE WARFIGHTER'S SIMULATION 2000 DEVELOPMENT TEAM USED TO DEVELOP THE REQUIREMENTS DOCUMENT FOR THE ARMY'S NEXT GENERATION OF MILITARY SIMULATIONS.				
14. SUBJECT TERMS  WARFIGHTER'S SIMULATION 2000 TRAINING REQUIREMENTS ANALYSIS PROGRAM			15. NUMBER OF PAGES 124	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT UNCLASSIFIED	18. SECURITY CLASSIFICATION OF THIS PAGE UNCLASSIFIED	19. SECURITY CLASSIFICATION OF ABSTRACT UNCLASSIFIED	20. LIMITATION OF ABSTRACT	

# **WARFIGHTER'S SIMULATION 2000 TRAINING REQUIREMENTS ANALYSIS PROGRAM**

**Major Robert G. Phelan Jr.  
Captain Michael D. McNett  
Lieutenant Colonel Michael L. McGinnis**

**A JOINT TECHNICAL REPORT  
OF THE  
OPERATIONS RESEARCH CENTER  
and  
DEPARTMENT OF ELECTRICAL ENGINEERING AND COMPUTER SCIENCE  
UNITED STATES MILITARY ACADEMY**

**Directed by  
Lieutenant Colonel Michael L. McGinnis, Ph.D.  
Director, Operations Research Center**

**Approved by  
Colonel James L. Kays, Ph.D.  
Professor and Head  
Department of Systems Engineering**

**The Operations Research Center is supported by the Assistant Secretary of the Army for Financial Management & Comptroller. The Sponsor for this project is the U.S. Army Simulation, Training and Instrumentation Command (STRICOM), Orlando, Florida, and the National Simulation Center, Fort Leavenworth, Kansas.**

## ACKNOWLEDGMENTS

This research was supported and funded by the U.S. Army Simulation, Training and Instrumentation Command and the National Simulation Center. The authors are especially thankful for advice and contributions to this project made by the following individuals: Colonel (ret.) Bob White of the Institute for Defense Analysis, Lieutenant Colonel George Stone of the National Simulation Center, Major Frank Rhinesmith of STRICOM and Mr Dan French of Veda, Inc.

## TABLE OF CONTENTS

ACKNOWLEDGMENTS .....	ii
TABLE OF CONTENTS.....	iii
TABLE OF FIGURES.....	v
LIST OF TABLES.....	vii
1.0 INTRODUCTION.....	1
2.0 STATEMENT OF WORK .....	2
3.0 DEFINING THE T-RAP.....	4
4.0 PROGRAMMING ON THE WORLD WIDE WEB.....	8
5.0 SYSTEM IMPLEMENTATION .....	9
6.0 DATABASE AND USER INTERFACE DESIGN.....	14
7.0 OUTPUT ANALYSIS .....	32
8.0 LESSONS LEARNED.....	36
9.0 CONCLUSIONS AND FUTURE WORK .....	37
10.0 REFERENCES .....	38

APPENDIX A - SUPPORTED MISSION TRAINING PLANS.....	39
APPENDIX B - MTP CONVERSION ANNOTATED SOURCE CODE.....	48
APPENDIX C - SOURCE CODE FOR CREATION OF USER ACCOUNTS (AUTH.EXE) .....	55
APPENDIX D - SOURCE CODE FOR DISPLAY OF TRAINING AUDIENCE TASKS (MAPPING.EXE) .....	60
APPENDIX E - SOURCE CODE FOR DISPLAY OF TRAINING AUDIENCE TASKS (UPDTPTA.EXE) .....	66
APPENDIX F - SOURCE CODE FOR DISPLAY AND SELECTION OF REPRESENTED UNIT TASKS (RUXS.EXE).....	71
APPENDIX G - SOURCE CODE FOR EVALUATION OF REPRESENTED UNIT TASKS (MAPRU.EXE) .....	90
APPENDIX H - SOURCE CODE FOR DETAILED SELECTION OF REPRESENTED UNIT TASKS (MAPRU2.EXE).....	94
APPENDIX I - SOURCE CODE FOR PRESENTATION OF REPRESENTED UNIT TASK DETAILS (UPDTRU.EXE) .....	98
APPENDIX J - SOURCE CODE TO BUILD QUERY ENGINE (QUERY.EXE).....	105
APPENDIX K - SOURCE CODE FOR EXECUTION OF SELECTED QUERY (RUNQUERY.EXE) .....	110
APPENDIX L - WEIGHTING OF SME RATINGS (SME <sub>F</sub> ).....	115

## TABLE OF FIGURES

Figure 1. Task Proficiency Support Code Logic.....	5
Figure 2. Criteria Used for TPSC Evaluations.....	5
Figure 3. Roll-up Logic - Standard to Subtask .....	6
Figure 4. Roll-up Logic - Subtask to Task.....	6
Figure 5. Represented Unit Rating Criteria. ....	7
Figure 6. Interfacing a Program with a World Wide Web User using CGI.....	9
Figure 7. User Navigation.....	16
Figure 8. Collecting User Information - Upper Half of Screen. ....	17
Figure 9. Collecting User Information - Lower Half of Screen.....	18
Figure 10. Selection of Training Audience.....	20
Figure 11. Selection of Training Audience Tasks.....	21
Figure 12. Selection of Training Audience Tasks.....	24
Figure 13. Evaluation of Training Audience Tasks. ....	25
Figure 14. Evaluation of Training Audience Tasks (Continued).....	26
Figure 15. Represented Unit Status Screen.....	28
Figure 16. Selection of Represented Unit Tasks Screen.....	29
Figure 17. Selection of Represented Unit Tasks Screen.....	30
Figure 18. Evaluation of Represented Unit Tasks Screen.....	31
Figure 19. Standard Query Screen .....	34

Figure L-1: Concept for Weighting SME Ratings .....	115
Figure L-2. Relative Value Factors Based on Position Experience ( $V_x$ ) .....	118
Figure L-3. Contribution Factors Based on Type of Experience ( $X_i$ ) .....	119
Figure L-4. Contribution Factors Based on Echelon of Experience ( $Z_k$ ) .....	119
Figure L-3: Sample Experience Matrix .....	122

## LIST OF TABLES

Table 1. Sample R-TASK MTP Listing.....	11
Table 2. Sample Task Condition Standard Listing after conversion. ....	13
Table 3. Task Steps and Performance Measures Extract. ....	14
Table 4. Extract of Error Trapping Database.....	19
Table 5. Extract of Selection of Training Audience Tasks Database Table .....	22
Table 6. Extract of Selection of Training Audience Tasks Database Table .....	27
Table 7. Extract of Linkage of Training Audience Task to Represented Unit.....	27
Table 8. Extract of Linkage of Training Audience Task to Represented Unit Task.....	32
Table 9. Extract of Represented Unit Task Detailed Data.....	32
Table L-1. F <sub>SA</sub> Factors .....	116
Table L-2. Normalized SME Multipliers.....	120
Table L-3. TPSC Heuristic Algorithm.....	120
Table L-4. TSSC Heuristic Algorithm.....	121

## EXECUTIVE SUMMARY

The World Wide Web (WWW) enables users to present and collect information from others around the world. It provides users with access to information on any subject using standard browsers.

This technical report discusses the design and implementation of a multi-user database system called the Training Requirements Analysis Program (T-RAP), on the World Wide Web that the Warfighter's Simulation (WARSIM) 2000 development team used to develop the requirements document for the Army's next generation of military simulations. A total of 262 U.S. Army Mission Training Plans (MTP) representing units from Platoon to Corps have been converted into a database format and currently available for use by authorized users. The system converts database entries to Hyper-Text Markup Language (HTML) through the use of the Common Gateway Interface (CGI) commonly used by most WWW servers. These MTPs are provided to users to gather their opinions about the suitability of training each task within the WARSIM system. They are also useful for determining which computer generated force tasks must be coded to support unit training. The system stores and links individual user opinions with each MTP task for every step and performance measure associated with that task. The system also makes previously entered opinions available to users every time they log on to the system.

The Standard Query Language enables users to analyze, combine and display data. The system features over 20 standard queries used by the WARSIM 2000 developers in ascertaining which MTP tasks to focus on during the software development phase of the simulation's acquisition.

## 1.0 INTRODUCTION

According to the National Simulation Center, "WARSIM 2000 will provide simulation tools to Force XXI leaders which they can use to create realistic operational conditions for education, training, and mission rehearsal. The program objectives include supporting Total Army and Joint Force events from battalion through echelons above corps in scenarios from across the operational continuum while reducing the resources required to prepare, execute, and assess simulation events. WARSIM 2000 will support real-time battle command training events such as seminars, command post exercises (CPXs) and Battle Command Training Program (BCTP) events in all type units and schools."<sup>1</sup>

WARSIM 2000 is under development to support training of commanders and staffs at battalion through echelons above corps level. It will be able to support single or multi-echeloned training with both actual and synthetic units. The tasks that WARSIM will support are primarily command and staff tasks which are cognitive and not physical in nature. These tasks include planning, coordinating and directing operations of forces under the control of the training staff or audience as well as functioning as a subordinate element to other headquarters elements. The tasks are primarily information oriented and may generate information products, updates or transmit a decision.

In October 1996, MAJ(P) George Stone, National Simulation Center (N.S.C.), with duty at the U.S. Army Simulation Training and Instrumentation Command, (STRICOM) approached the Operations Research Center, United States Military Academy to ask for help in developing the Training Requirements Analysis Program (T-

RAP). This technical report describes the development of the World Wide Web based T-RAP collection tool for the N.S.C. and STRICOM. It documents the rationale and detail of the functions, form and data structures used throughout the system. The report also discusses the development of the on-line data analysis function through which users can execute one of a variety of pre-defined queries specific to their analysis needs.

The T-RAP supports the WARSIM 2000 requirements development process by providing a means with which specific staff training events and tasks are analyzed by Subject Matter Experts (SMEs) to evaluate their supportability within the WARSIM system. These events and tasks include both the tasks that an actual training unit would perform while using the WARSIM system as well as those tasks that computer generated forces would perform to support/ stimulate the training unit in accomplishing any particular task. The T-RAP also provides the WARSIM developers the opportunity to analyze a fine level of detail about all aspects of the training audience tasks as well as any simulated forces tasks.

## 2.0 STATEMENT OF WORK

Discussions with NSC and STRICOM and the Operations Research Center submitted led to the following statement of work submitted by the ORCEN:

---

<sup>1</sup>[http://www-leav.army.mil/nsc/warsim/ord\\_3\\_1.htm](http://www-leav.army.mil/nsc/warsim/ord_3_1.htm)

Statement of Work for the Development of a WARSIM 2000 Requirements Documentation System

The Operations Research Center of Excellence, (ORCEN), United States Military Academy, will develop a *Subject Matter Expert Data Collection and Analysis Tool*. The fully developed tool will be used by WARSIM 2000 Product Managers to generate a complete Requirements Document by 1 March 1997.

Requirement:

The Operations Research Center, The U. S. Army Simulation, Training and Instrumentation Command (STRICOM), Orlando Florida, and the National Simulation Center (NSC), Fort Leavenworth, Kansas, propose to jointly develop a general methodology for generating database and modeling requirements for WARSIM 2000. This approach links actual training units using WARSIM 2000 with supporting requirements. These requirements include but are not limited to the following: Synthetic Represented Units, Synthetic Environment Objects, and additional unit equipment that must be available for WARSIM training. Besides identifying training unit support requirements, this effort will also determine the applicability and relative importance and of unit training tasks and supporting requirements. Determining relative importance will enable the developers to prioritize efforts of the WARSIM contractor (Lockheed Martin Information Systems) during software development. The development team will also construct an information repository for recording knowledge from Subject Matter Experts across the country thus eliminating the need for travel. Finally, all collected data will be fully integrated into WARSIM's Functional Description of the Battlespace (FDB).

Technical Approach

The ORCEN shall perform the following tasks to support the WARSIM 2000 requirements development enumerated above:

- 1) Develop a relational database for data collection and analysis using the previously defined general approach (see above). The approach will also ensure complete traceability exists of information input into the data repository.
- 2) Test the basic approach using multiple experts. Currently, the algorithm determines the relative task importance based upon the opinions of one expert. Obviously, as the system matures, it must be able to receive and handle multiple opinions from experts.
- 3) Develop a methodology for prioritizing synthetic units and supporting tasks relating to Primary Training Audience training tasks.
- 4) Wherever possible, commercial-off-the-shelf (COTS) software and existing technologies will be used to develop a World Wide Web (WWW) site using Common Gateway Interface (CGI) and relation database techniques to extract opinion from the various Subject Matter Experts.
- 5) Allow users to use standard WWW browsers as their input tool.
- 6) Develop and articulate data dependencies to allow incorporation of collected data into the FDB.
- 7) Define clear linkages from approved Army documents to data presented or used by the Requirements Documentation System.
- 8) Transfer the system from the ORCEN to STRICOM, Orlando for permanent use and maintenance.

Timelines:

- 1) 28 October - 9 December: SYSTEM DEVELOPMENT. Closely coordinate with STRICOM, NSC, and contractors during tool development.
- 2) 9 December: SYSTEM DEMONSTRATION AND AVAILABILITY. Data structures must be compliant and integratable with the FDB.
- 3) 9 December 1996 - 15 January 1997: DATA COLLECTION. The ORCEN will be available to solve any system problems that may develop.
- 4) 15 January - 15 February: DATA ANALYSIS AND REPORT GENERATION.
- 5) February 1997: TECHNICAL REPORT. Present findings to STRICOM and NSC. Submit report and provide Product Managers with complete system documentation.

Financial Support: Details of the financial support provided to the Operations Research Center are not provided in this report.

### 3.0 DEFINING THE T-RAP

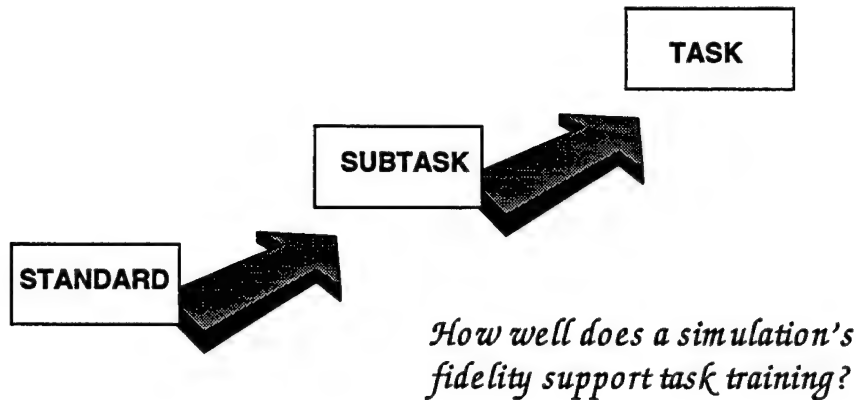
The key functions of the T-RAP as defined by STRICOM and NSC are:

- Collect Data
- Perform Data Analysis
- Provide Feedback to Users
- Provide Summary to the WARSIM Development team.

**Collect Data:** The T-RAP is designed to identify the training units with related tasks and equipment that will be supported by WARSIM 2000. It also links Represented Units (RU) or Computer Generated Forces and the tasks they will accomplish within the simulation to each supported training unit task. It provides the opportunity to describe any synthetic environment objects as well as significant pieces of equipment that must be present within the simulation to support either training or represented unit tasks.

**Perform Data Analysis:** Task Performance Support (TPS) codes provide a relative indicator of the degree to which a given training simulation supports the performance of collective training tasks. Any collective task that is to be trained on any simulation system in the Army will have a TPS code associated with it. The code provides commanders with the ability to discriminate between simulation training events and decide which simulation platform better supports his training goals. STRICOM and NSC desired to compute the TPS code for each evaluated training audience task for the WARSIM 2000 system. Computation of the rating is accomplished through the use of the flow charts in Figures 1 through 4 which were provided by STRICOM and were previously used during the Close Combat Tactical Trainer requirements development process.

# MTP "TASK TRAINABILITY" ROLL-UP LOGIC



C:\msoffice\powerpnt\tradelrollup - 7 Feb 95

Figure 1. Task Proficiency Support Code Logic.

## CRITERIA for Rating MTP Standards

---

**Highly Supported (H):** The standard can be met entirely. Sufficient cues are present within the environment and appropriate responses are supported such that the training experience is much the same as it would be in a field environment.

**Moderately Supported (M):** Most of the the standard is supported. Sufficient cues are present within the environment and appropriate responses are supported such that the training experience is sufficiently like a field training environment to support positive training of tactics and techniques.

**Outside Support Required (O):** The standard can be met in the training environment, but only if over half of the required cues and reponses are provided or performed with resources outside/adjacent to the environment.

**Not Supported (N):** The standard cannot be supported. A significant portion of the cues and responses required to perform the required actions are not represented or cannot be supported.

Figure 2. Criteria Used for TPSC Evaluations.

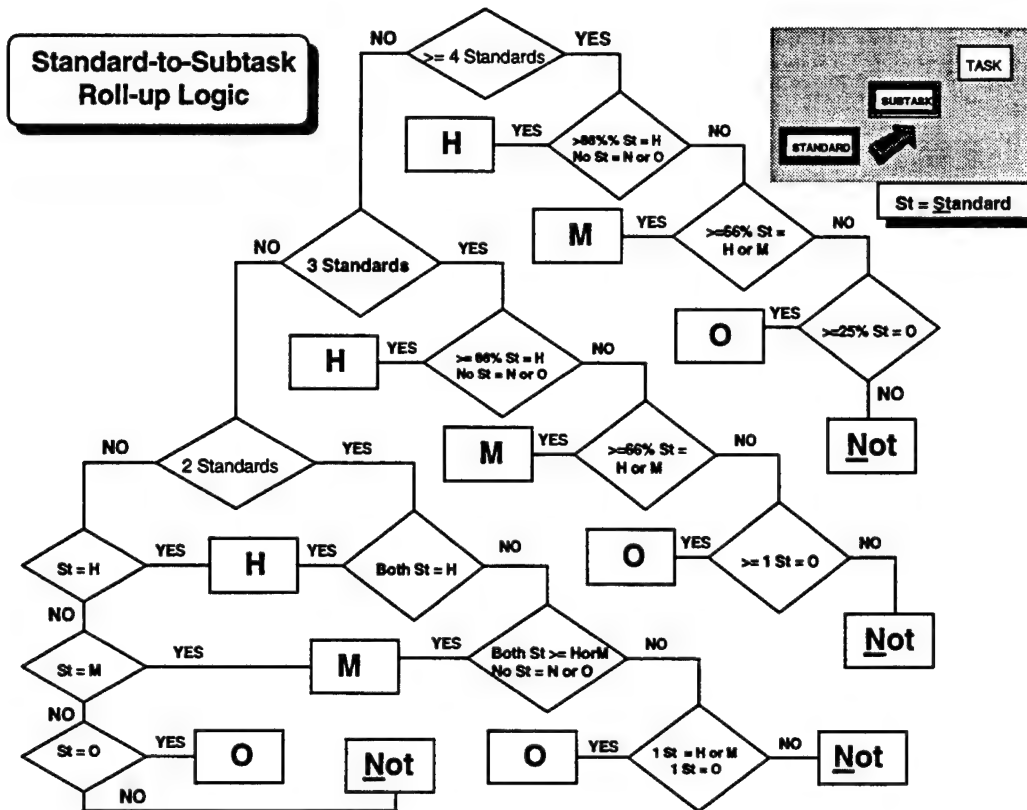


Figure 3. Roll-up Logic - Standard to Subtask

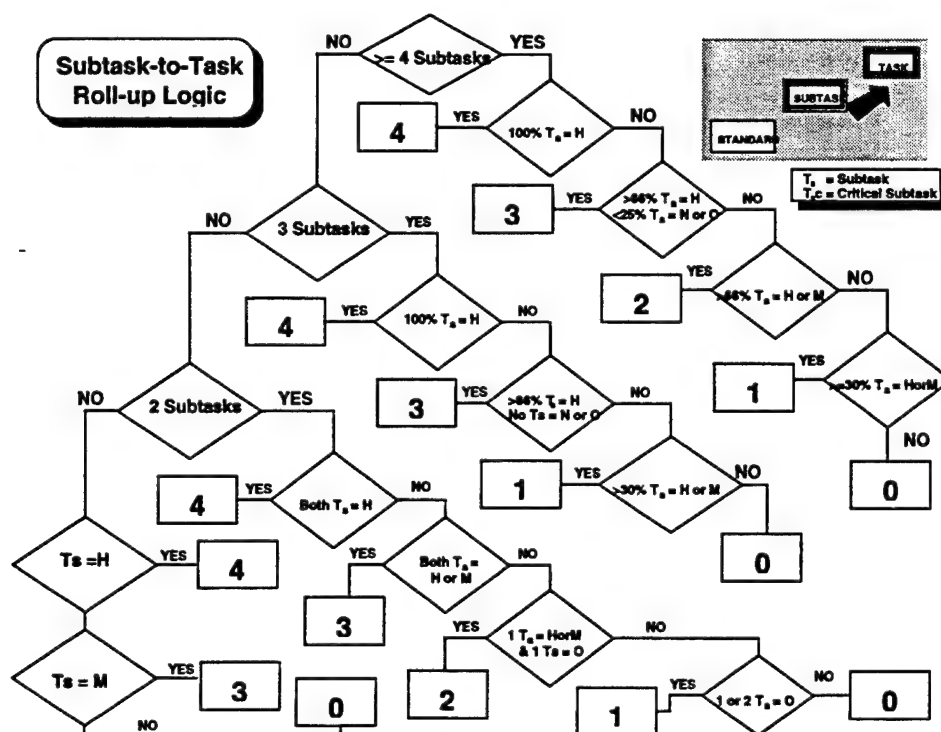


Figure 4. Roll-up Logic - Subtask to Task.

STRICOM also defined a new code for the T-RAP and the WARSIM development process, the Task Simulation Support Code (TSSC). The TSSC describes the degree of detail to which a Represented Unit task will be modeled in the simulation to stimulate training of a training audience task. This code then characterizes a Represented Unit's task as to the level of fidelity required within the simulation. STRICOM directed that the same computational scheme as the TPSC be used for the TSSC. Since the expert would now be asked to rate the task as to the level of fidelity with which it should be modeled within the simulation, the criteria were changed as shown in Figure 5.

### **CRITERIA**

#### **for Rating MTP Standards for Represented Units**

---

**Full Details (H):** All aspects of this RU's TS/ PM must be modeled in the simulation to stimulate the associated TA task.,

**Partial Details (M):** Only essential aspects of this RU's TS/ PM must be modeled in the simulation to stimulate the associated TA task.

**Minimum Details (O):** All that is required is to know that the RU TS/PM has been executed (Yes/NO?).

**Not Required (N):** No information required.

**Figure 5. Represented Unit Rating Criteria.**

STRICOM also directed that the T-RAP collect SME opinions about the synthetic environment and unit equipment within the WARSIM system and required by represented units. Synthetic environment objects are any objects required in the simulation that are needed to execute the task step/ performance measure of the associated represented unit

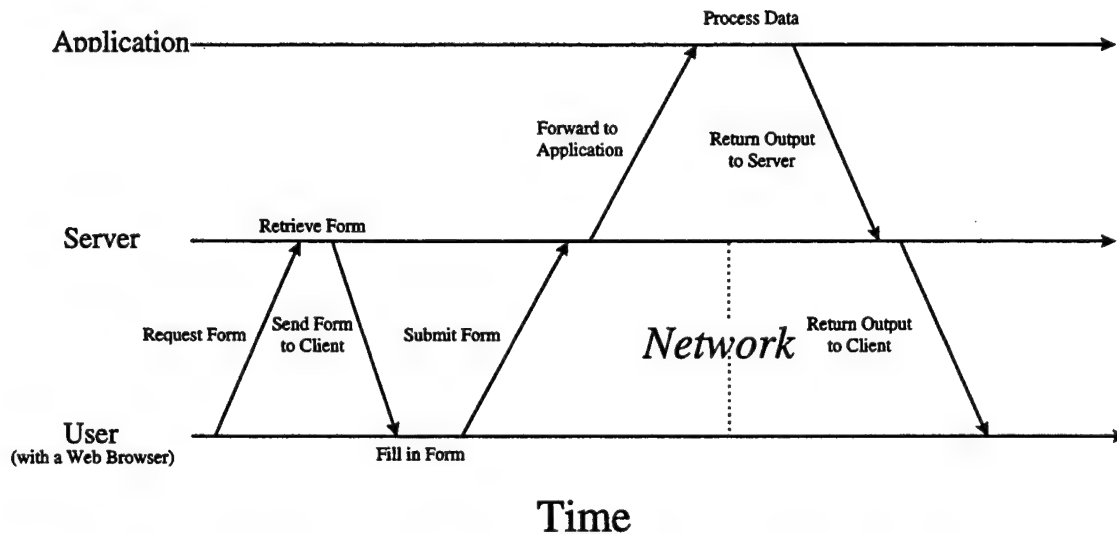
task. Unit equipment is defined as any major end items that are required to execute the task step/ performance measure of the represented unit task.

#### 4.0 PROGRAMMING ON THE WORLD WIDE WEB

STRICOM initially developed a Microsoft Excel version of the T-RAP. This version was capable of collecting a single user's opinions to characterize both training audience and represented unit tasks. They quickly realized that there would be a tremendous problem in merging data from all of the individual spreadsheets once collected from SMEs across the Army. They also realized that that the spreadsheet suffered from 'version control' problems, meaning that as their tool developed, they were unsure who had what version. As the functionality of their tool improved, they had users with different versions of the tool. To solve the data collation problem, a database approach had to be implemented so that multiple expert opinions could be collected and analyzed. The World Wide Web as the collection medium was selected for a number of reasons. First of all, once implemented, users from all over the world could use the tool from their home station thus eliminating any travel related expenses. Also, a program on the web would correct the version control problem because as the system continued in its development process, users would have access to the latest and most current version on-line.

Linking a database to a world wide web server requires then a program which can execute necessary routines on the WWW server and return results to a user over the network. The Common Gateway Interface (CGI) is part of a Web Server that can communicate with other programs running on the server. In CGI, the Web server can call

up a program and pass user-specific data to the program. The application program then processes the data and passes the program's response back to the Web browser. Figure 6 depicts the interaction between a user on a network with a standard browser inputting and receiving processed data from a web server.



**Figure 6. Interfacing a Program with a World Wide Web User using CGI.**

## 5.0 SYSTEM IMPLEMENTATION

The first step in designing the on-line version of the T-RAP was to select both a programming language and database product that were compatible as well as capable of generating CGI scripts. Microsoft® Visual Basic® was chosen as the CGI programming environment because of the authors' experience with the product on previous ORCEN projects. To maintain software consistency and transferability, Microsoft® Access® was also selected as the underlying multi-user database engine. Given the relatively short development time of the T-RAP, it was decided to go with these products to quickly program the prototype. There are more robust database and programming products

available in the market today and future versions of the T-RAP may want to explore other software options.

The initial step in implementing the T-RAP was to translate all applicable Mission Training Plans (MTPs) into a format which Access® would recognize and use. Currently the U.S. Army Training Support Center, Headquarters TRADOC, is responsible for maintaining the *Automated Systems Approach to Training*. This product contains the complete documentation of all approved MTPs for the Army. The MTPs are organized into a database format using Infomaker® for Windows by Powersoft®. After several weeks of attempting to translate the MTPs from this format into an Access® compatible one that the CGI environment could use, the authors decided to seek another MTP source.

LTC Stone then recommended that we contact Resource Consultants Incorporated (RCI) of Orlando, Florida who had performed some previous work for STRICOM supporting the TREDs and CCTT programs. They had developed a CDROM called R-TASK that had every MTP in Microsoft® FoxPro® format. Table 1 provides an extract of one of the MTPs contained within that product. Looking at the table, it is obvious that the database actually contains every line and sentence within the MTP identified as unique records and some code would be required to convert the data for T-RAP use.

**Table 1. Sample R-TASK MTP Listing.**

DOCKEY	SECTION	TAGKEY	LEVEL	PAR	LINE	TEXTTYPE	TEXT
0000000001	%	022-8-CG01	2	1	1	BOS	ARTEP 100-15-MTP MISSION TRAINING PLAN FOR CORPS COMMAND GROUP AND STAFF
0000000001	%	022-8-CG01	2	1	2	BOS	(1990/05/01)
0000000001	%I	022-8-CG01	2	1	1	BOS	BOS: COMMAND AND CONTROL
0000000001	%I	022-8-CG01	2	2	1	BOS	ELEMENT: COMMANDING GENERAL, COMMAND GROUP
0000000001	%II	022-8-CG01	3	1	1	TSK	TASK: COMMAND AND CONTROL CORPS OPERATIONS (FM 100-5, FM 100-15, FM 101-5)
0000000001	%II	022-8-CG01	3	1	2	TSK	(022-8-CG01)
0000000001	%II	022-8-CG01	3	1	3	TSK	
0000000001	%II	022-8-CG01	3	2	1	CND	CONDITIONS:
0000000001	%II	022-8-CG01	3	2	2	CND	The corps, as part of a Theater/Army operating in a joint and combined
0000000001	%II	022-8-CG01	3	2	3	CND	environment, is engaged in combat or combat is imminent. The corps' tactical,
0000000001	%II	022-8-CG01	3	2	4	CND	main and rear command posts have been deployed and established. Communications
0000000001	%II	022-8-CG01	3	2	5	CND	have been established and reports are being received from subordinate units and
0000000001	%II	022-8-CG01	3	2	6	CND	submitted to Theater/Army headquarters in accordance with tactical standing
0000000001	%II	022-8-CG01	3	2	7	CND	operating procedures (SOPs). Combat may involve offensive, defensive or
0000000001	%II	022-8-CG01	3	2	8	CND	retrograde operations. Continuous operations (day and night) are envisioned.
0000000001	%II	022-8-CG01	3	2	9	CND	Enemy forces are composed of armored, mechanized and airborne forces with their
0000000001	%II	022-8-CG01	3	2	10	CND	associated combat support, combat service support and air assets. The NBC
0000000001	%II	022-8-CG01	3	2	11	CND	environment is active. Friendly forces possess the capability to employ nuclear
0000000001	%II	022-8-CG01	3	2	12	CND	or chemical weapons. The enemy is employing chemicals in support of terrain
0000000001	%II	022-8-CG01	3	2	13	CND	denial operations and against selected deep targets. They have nuclear weapons

The next step was to develop a method with which to translate the R-TASK MTP databases into Access® along with an accompanying schema that would lend itself to the data collection effort. In writing the translation code, the Tasks, Conditions and Standards for each MTP were separated from the individual steps and performance measures associated with each of the tasks. During that phase of the T-RAP

development, the concern was to be able to separate the MTP into its component parts to maintain satisfactory system performance as each of the components would be used for different aspects of the collection effort. Table 2 provides an extract of the resulting Task, Condition, Standard table after conversion. What is significant in this table is that every task is sequentially indexed. This indexing provides that basis to link individual user opinions to individual tasks.

**Table 2. Sample Task Condition Standard Listing after conversion.**

BOS	Tasknumber	Task	Condition	Standard	Counter
COMMAND AND CONTROL	022-8-CG01	COMMAND AND CONTROL CORPS OPERATIONS (FM 100-5, FM 100-15, FM 101-5) (022-8-CG01)	The corps, as part of a Theater/Army operating in a joint and combined environment, is engaged in combat or combat is imminent. The corps' tactical, main and rear command posts have been deployed and established. Communications have been established and reports are being received from subordinate units and submitted to Theater/Army headquarters in accordance with tactical standing operating procedures (SOPs). Combat may involve offensive, defensive or retrograde operations. Continuous operations (day and night) are envisioned. Enemy forces are composed of armored, mechanized and airborne forces with their associated combat support, combat service support and air assets. The NBC environment is active. Friendly forces possess the capability to employ nuclear or chemical weapons. The enemy is employing chemicals in support of terrain denial operations and against selected deep targets. They have nuclear weapons in a ready posture but have not yet employed them. Their biological weapons remain in storage at Front level. Further use of chemical or initial employment of nuclear or biological weapons may occur at any time. Electronic warfare (EW) environment is active. The enemy is employing radio electronic combat (REC) tactics. There is air parity within the theater of operations; however local air superiority can be achieved for short periods of time by either air force. Conflict is characterized by nonlinear operations; a wide range of enemy weapon systems whose ranges and lethality equal or exceed U.S. capabilities; and numerous surveillance, target acquisition and communication sensors. The combat environment is characterized by a continuing series of fast, chaotic, obscure battles. The corps has received the INTSUM and the Theater/Army OPLAN. The corps commander's intent is for the corps to accomplish its assigned mission while retaining the capability continue operations required.	The corps' mission is accomplished IAW the Theater/Army commander's intent. The resources are employed to effectively accomplish the mission. The corps retains the capability to continue combat operations.	1

In all, there were 262 MTPs available in R-TASK that were converted for use.

Appendix A lists all supported MTPs as of the date of this technical report. Table 3 provides an extract of the conversion of the steps and performance measures from the R-TASK database. These tables are linked to the original table in that the MTP Task

number listed in the table is the same as the Counter in the task, condition, standard table.

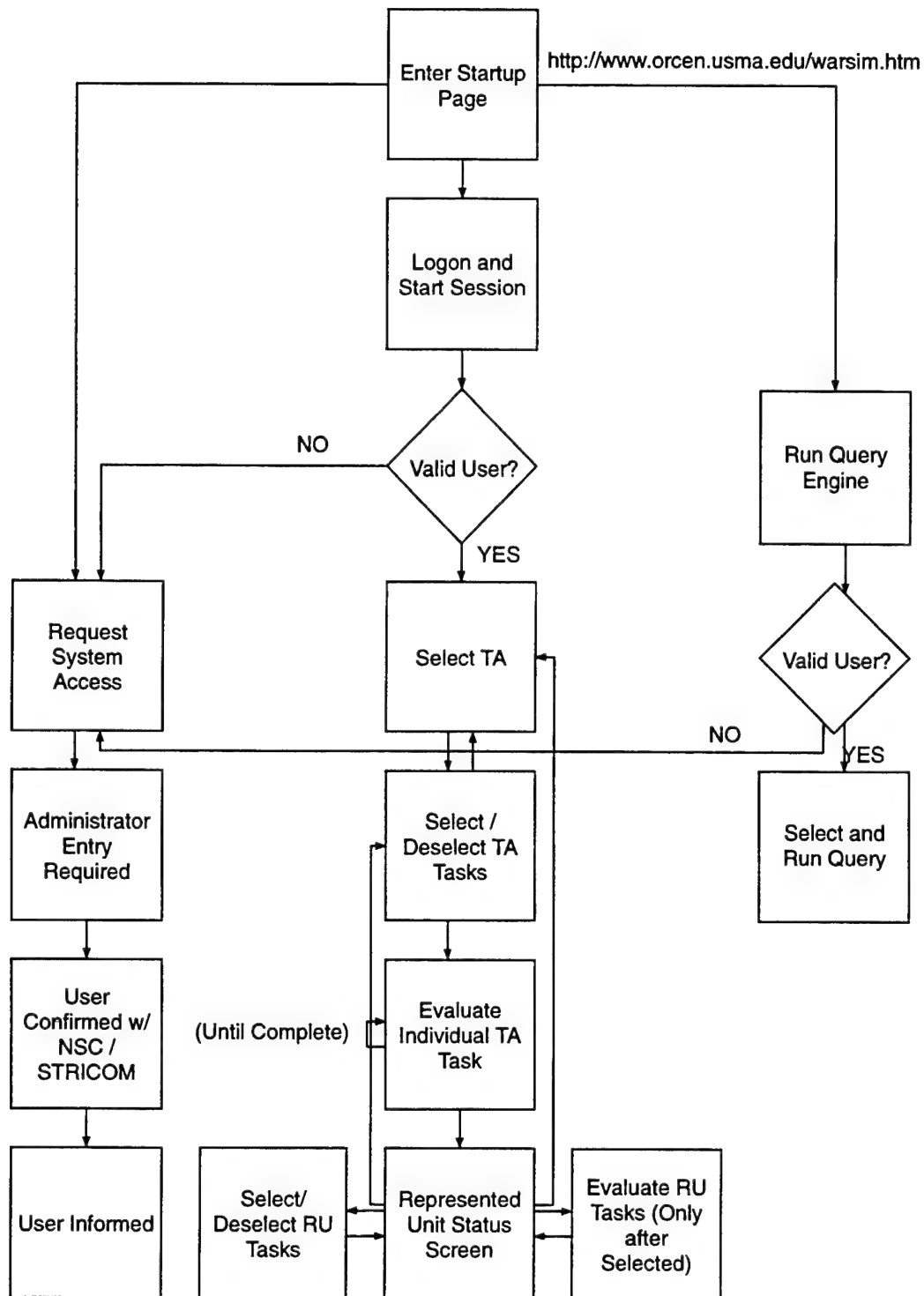
Appendix B provides the annotated Visual Basic ® source code for both of the conversion routines.

**Table 3. Task Steps and Performance Measures Extract.**

Tasknumber	Step	Narrative	MTPTaskReference	MTPStepReference	MTPPMReference	MTPSubPMReference
022-8-CG01	1	*1. Plans corps operations.	1	1	0	0
022-8-CG01	1 a	a. Provides information to staff.	1	1	1	0
022-8-CG01	1 b	b. Approves staff's restated mission.	1	1	2	0
022-8-CG01	1 c	c. Issues guidance to staff.	1	1	3	0
022-8-CG01	1 d	d. Makes the commander's estimate.	1	1	4	0
022-8-CG01	1 e	e. Clearly articulates his intent and concept of the operation.	1	1	5	0
022-8-CG01	2	*2. Organizes corps resources.	1	2	0	0
022-8-CG01	2 a	a. Allocates resources to support concepts of operation.	1	2	1	0
022-8-CG01	2 b	b. Task organizes corps resources.	1	2	2	0
022-8-CG01	2 c	c. Establishes priorities of corps resources.	1	2	3	0
022-8-CG01	2 d	d. Establishes procedures and relationships.	1	2	4	0
022-8-CG01	2 e	e. Modifies established procedures, as required.	1	2	5	0
022-8-CG01	2 f	f. Establishes procedures for receiving major non-organic unit.	1	2	6	0
022-8-CG01	2 g	g. Designates, sustains and shifts, as necessary, the main effort.	1	2	7	0
022-8-CG01	2 h	h. Establishes procedures to eliminate C2 degradation during shift changes.	1	2	8	0
022-8-CG01	3	*3. Directs corps operations.	1	3	0	0

## 6.0 DATABASE AND USER INTERFACE DESIGN

The following section highlights the development of the user interface and underlying database structure. The included functions closely parallel the spreadsheet model provided by STRICOM at the beginning of system development. Figure 7 graphically portrays the built-in navigation logic for an user. As the system developed, it was clear that given the variety of screens that were under development, T-RAP navigation should be built in, thus allowing an user to provide his opinions with little risk of becoming *lost*. Each aspect of the folw chart is discussed in great detail for the remainder of this section.



**Figure 7. User Navigation.**

**REQUEST SYSTEM ACCESS:** A new user entering the system is only allowed to proceed to the *Request System Access* function. He is presented with a Hyper-Text Markup Language (HTML) form that requires him to fill in identifying data.

**WARSIM 2000 User Access Request Form**

You have been sent this sheet because you lack a valid userid/ password for the WARSIM 2000 Subject Matter Expert Opinion Form. If you believe you have a valid account, try again.

Name	<input type="text"/>
Title	<input type="text"/>
Organization	<input type="text"/>
Street Address	<input type="text"/>
Phone Number	<input type="text"/>
Email Address	<input type="text"/>
Requested Login Name:	<input type="text"/>
Requested Password:	<input type="text"/>
Password (Enter again to check):	<input type="text"/>
Subject Matter Expert Field(s) Use CTRL or SHIFT keys to select multiple fields	<input type="checkbox"/> Infantry / Armor <input type="checkbox"/> Field Artillery <input type="checkbox"/> Air Defense Artillery <input type="checkbox"/> Engineer <input type="checkbox"/> Aviation <input type="checkbox"/> Signal <input type="checkbox"/> Combat Service Support <input type="checkbox"/> Brigade Level Operations <input type="checkbox"/> Corps / Division Level Operations
Comments	<input type="text"/>

Indicate your Military Staff Experience in the following environments:

**Figure 8. Collecting User Information - Upper Half of Screen.**

At the same time, the system requests some of the experience this user has had while serving in the military as well as any simulation experience. This data is used in the analysis module of the system when multiple experts' opinions are weighed against one another.

Indicate your Military Staff Experience in the following environments:  
(as many as appropriate).

	Actual Deployment	JRTC	CMTC	NIC	BCTP	ECBST	Other
Commander	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>
XO	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>
S-1	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>
S-2	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>
S-3	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>
S-3 Air	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>
S-4	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>
FSO	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>
Staff Engineer	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>
Staff ADA	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>
Staff Chemical	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>
ALO	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>
SIGO	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>
Other	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>	Brigade: <input type="checkbox"/> Battalion: <input type="checkbox"/>

Figure 9. Collecting User Information - Lower Half of Screen.

Once the information is submitted, the data is written to an Access® database by the CGI system. Experience with managing the system has shown that most prospective users are eager to get on the system so they will send the administrators an email message requesting immediate access. The administrators contact NSC or STRICOM and seek verbal approval of users who are then given access to the web site itself and can then proceed into with data collection or analysis. The users already have their userid and password as they are stored in the database. Another CGI program was written to ensure that all users have unique userids, that they enter their passwords twice and that all required fields are filled out. If not, the user is presented with an error message and

directed to correct the deficiency. Appendix C provides the source code listing for the creation of user accounts.

**ERROR TRAPPING:** As the T-RAP developed, there were numerous instances early on where one of the programs may have *crashed* and the user provided some sort of error message. Rather than rely upon users providing the message text, an error trapping routine was developed where the actual error message that an user received was written to an Access® database. After examining the error code, the developers then attempted to duplicate the error condition. Once it was duplicated, it became then very easy to correct the deficiency within the software. Table 4 provides a sample listing of some of the received error codes. This system is still in place and provides a measure of the stability of the software. No errors have been noted in the operating code for several months.

**Table 4. Extract of Error Trapping Database**

User	ErrorString	DTG
phelan	Error in /wincgi/makeru.exe-Couldn't use 'c:\netscape\server\docs\input\database\mtptpsc.mdb'; file already in use. (error #3045)	12/12/96 8:35:07 AM
wam	Error in /wincgi/mapping.exe-Couldn't use 'c:\netscape\server\docs\input\database\mtptpsc.mdb'; file already in use. (error #3045)	12/12/96 8:42:20 AM
phelan	Error in /wincgi/makeru.exe-Couldn't use 'c:\netscape\server\docs\input\database\mtptpsc.mdb'; file already in use. (error #3045)	12/12/96 8:44:05 AM
wam	Error in /wincgi/updtpa.exe-Couldn't find input table or query ". (error #3078)	12/12/96 12:06:57 PM
wam	Error in /wincgi/ruxs.exe-Couldn't update; currently locked by user 'Admin' on machine 'ORCEN_SERVER'. (error #3260)	12/12/96 12:13:05 PM
phelan	Error in /wincgi/ruxs.exe-Couldn't update; currently locked by user 'Admin' on machine 'ORCEN_SERVER'. (error #3260)	12/12/96 1:36:21 PM
phelan	Error in /wincgi/ruxs.exe-Couldn't find input table or query ". (error #3078)	12/12/96 2:58:28 PM
phelan	Error in /wincgi/ruxs.exe-Couldn't find input table or query ". (error #3078)	12/12/96 2:58:46 PM

**SELECT TRAINING AUDIENCE:** Once a user receives approval and his account is entered into the web server, he selects the option to *Start a New Session*. After successfully logging in to the system, he is presented with a graphical representation of all of the available MTPs. This .gif file is coded as an *ismap* in HTML which maps the listed icons to their appropriate MTPs. If the user does not see the particular training

audience MTP he wishes to work with, he only has to click on any of the open space in order to receive a text listing of all available MTPs from the web server.

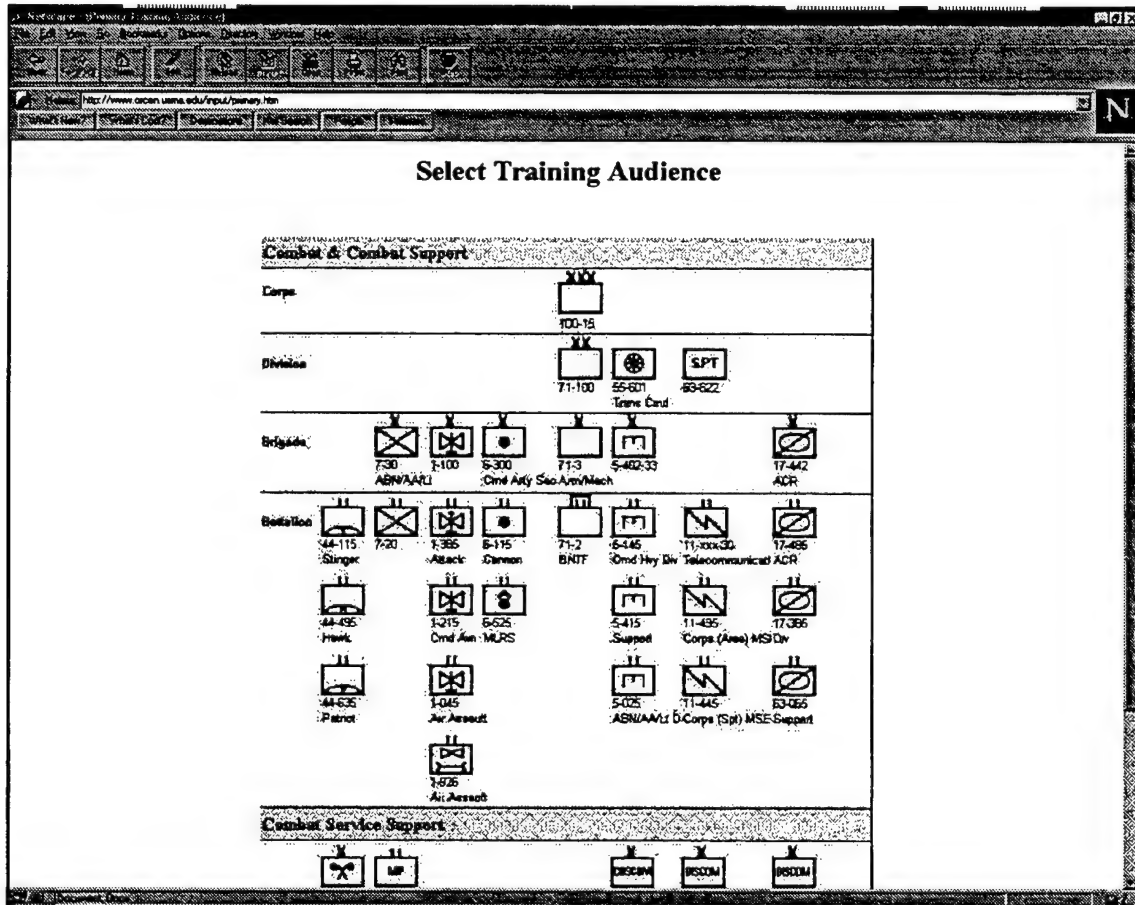


Figure 10. Selection of Training Audience.

**SELECT TRAINING AUDIENCE TASKS:** The next step in the evaluation process is the selection of those tasks which should be trained within WASIM 2000. Once the user selects any MTP, a CGI program is called that searches the database and extracts any records that match both the logged in user and his selected MTP. The T-RAP then dynamically begins to construct the HTML code and also places a check in any appropriate box indicating that the particular task has been previously selected as appropriate to train within WARSIM. Appendix D provides a listing of the Visual Basic source code that performs this function. These selected tasks are stored in a read/ write

Access® database. Table 5 depicts how the table is written within the system. Note from this table that the task is indexed by both user, MTP and task number. The update column indicates whether or not the user has evaluated that particular task and is used to support navigation between the various modules of the T-RAP.

COMMAND GROUP AND STAFF, HEAVY BRIGADE (71-3)

Select all of the Training tasks that will be trained within WARSIM.  
When complete, press the Submit button at the bottom of this screen.

Task Number	Task Conditions Standards	PTA TASK
	ELEMENT: ELEMENT: BRIGADE MI	
71-6-0018	TASK: EMPLOY MILITARY INTELLIGENCE COMPANY ASSETS	<input checked="" type="checkbox"/>
	ELEMENT: ELEMENT: COORDINATING STAFF	
71-6-0019	TASK: PLAN RECONNAISSANCE AND SURVEILLANCE	<input checked="" type="checkbox"/>
	ELEMENT: ELEMENT: BRIGADE S2 SECTION	
71-6-0077	TASK: PROVIDE INTELLIGENCE INPUT TO THE COMMAND ESTIMATE	<input checked="" type="checkbox"/>
	ELEMENT: ELEMENT: HHC COMMANDANT	
71-6-0246	TASK: PLAN AND SUPERVISE MAIN COMMAND POST SECURITY	<input checked="" type="checkbox"/>
	ELEMENT: ELEMENT: BRIGADE S2 SECTION	
71-6-0811	TASK: CONDUCT INTELLIGENCE FUNCTIONS FOR DEPLOYMENT	<input checked="" type="checkbox"/>
	ELEMENT: ELEMENT: BRIGADE TOC, TAC, REAR CP	
71-6-1080	TASK: EMPLOY OPERATIONS SECURITY MEASURES	<input checked="" type="checkbox"/>
	ELEMENT: ELEMENT: BRIGADE PROVOST MARSHAL	
71-6-1208	TASK: PLAN SECURITY OF DESIGNATED PERSONS, UNITS, CONVOYS, FACILITIES, AND MSR CRITICAL POINTS	<input checked="" type="checkbox"/>
	ELEMENT: ELEMENT: BRIGADE S2 SECTION	
71-6-2001	TASK: PREPARE THE INTELLIGENCE ESTIMATE	<input type="checkbox"/>
71-6-2002	TASK: PREPARE THE INTELLIGENCE ANNEX TO THE OPERATION ORDER	<input type="checkbox"/>
71-6-2003	TASK: ANALYZE INCOMING INFORMATION FROM MANEUVER ELEMENTS IN CONJUNCTION WITH INTELLIGENCE RECEIVED FROM HIGHER HEADQUARTERS G2	<input type="checkbox"/>
71-6-2004	TASK: MANAGE THE INTELLIGENCE EFFORT	<input type="checkbox"/>
71-6-2005	TASK: PROCESS SPECIFIC INFORMATION REQUIREMENTS DATA	<input type="checkbox"/>
71-6-2006	TASK: PROCESS COMBAT INFORMATION AND INTELLIGENCE	<input type="checkbox"/>
71-6-2007	TASK: CONDUCT AERIAL INTELLIGENCE COLLECTION SUPPORT PLANNING	<input type="checkbox"/>
71-6-2008	TASK: MAINTAIN THE BRIGADE INTELLIGENCE DATA BASE	<input type="checkbox"/>

Figure 11. Selection of Training Audience Tasks.

**Table 5. Extract of Selection of Training Audience Tasks Database Table**

Userld	Session	MTP	Tasknumbe r	Update	Rating	COMMENTS
rude	24	DMTP0001	12	yes	0	WARSIM must replicate enemy activity to the fullest extent to maximize training in a peacetime scenario.
rude	24	DMTP0001	8	yes	0	
rude	24	DMTP0001	7	yes	0	Replacement operations is contingent upon accurate personnel status reporting from subordinate echelons and consistent dialogue with higher Hqs S-1 to ensure replacement needs are identified and addressed.
rude	24	DMTP0001	6	yes	0	Requires consistent interaction with subordinate commanders and/or company TOCs thru personnel status reports.
rude	24	DMTP0001	2	yes	0	XO requires consistent, accurate and timely information from higher and subordinate elements
_rude	24	DMTP0001	1	yes	0	Successful execution of tasks require accurate & timely reporting from his subordinate elements

**EVALUATION OF INDIVIDUAL TRAINING AUDIENCE TASKS:** Once a set of tasks has been selected by a SME, then next step is to evaluate each task at the task step/ performance measure level. The user proceeds past the selection screen and is presented with a subsequent screen that provides a listing of all of the selected tasks and determines which have been previously evaluated and which have yet to be. The user is now forced to evaluate all selected tasks before being allowed to proceed with further

evaluation. In Table 5, a task annotated as Yes is considered evaluated. As a task is evaluated, the browser screen is dynamically updated and the user is provided some feedback as to his responses.

Evaluating each task is initiated by selecting one of the buttons presented on the navigation screen. The T-RAP then sends the user a detailed listing of the selected task to include the task, conditions and standards and all task steps and performance measures. Figures 13 and 14 depict portions of one brigade task, Employ Military Intelligence Company Assets. There are several items that an user must evaluate for this task. The first is to rate how well the WARSIM 2000 simulation will support each of the individual tasks and performance measures for the particular task. The standards that the user uses to rate the task were previously presented in Figure 2. At the same time, the WARSIM developers are also interested in collecting any user comments as to why they may have rated a particular step low. On the right hand side of the screen, the user is also asked to provide some level of detail on the communications methods and products required to support this task. At the bottom of the screen (Figure 13), the user is then asked to provide any task level type comments and to begin to link all Represented or Computer Generated Forces that are required to support this task. Appendix E provides the source code used to populate the screen with the task data and any previously entered user opinions.

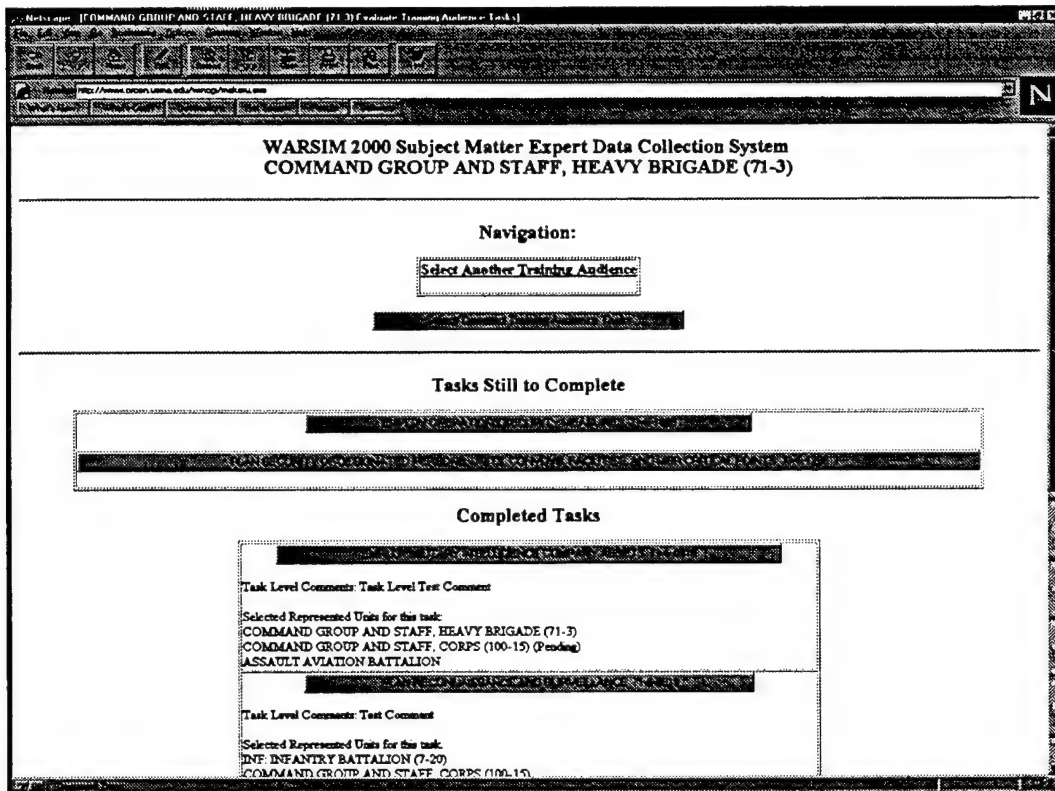


Figure 12. Selection of Training Audience Tasks.

Rate Training Audience Task - EMPLOY MILITARY INTELLIGENCE COMPANY ASSETS

Rate Steps and Performance Measures for COMMAND GROUP AND STAFF, HEAVY BRIGADE (71-3) performing  
EMPLOY MILITARY INTELLIGENCE COMPANY ASSETS

<p><b>TASK NUMBER:</b> 71-6-0018  <b>ELEMENT:</b> ELEMENT BRIGADE MI  <b>TASK:</b> EMPLOY MILITARY INTELLIGENCE  <b>COMPANY ASSETS (71-6-0018 KBMT):</b>  <b>CONDITIONS:</b> The brigade is preparing for or is engaged in combat. The brigade commander has designated the current task organization. The S2 is located at the main command post (CP). The main CP is operational. The S3 is developing or has issued an operation order (OPORD) or operation plan (OPLAN). Some iterations of this task should be performed in MOPP 4.  <b>STANDARDS:</b> The military intelligence (MI) company assets are fully used to support operations. Time required to plan and implement is increased when performing in MOPP 4.</p>	<p><b>TPSC TASK STEP PERFORMANCE MEASURE RATINGS</b></p> <p>Use the Blocks below to rate how well the simulation will support the individual Steps and Performance Measures of this Task.</p> <p><b>HIGHLY SUPPORTED (HIGH):</b> The TS/PM can be performed completely. Sufficient cues are present within the simulation environment and appropriate responses are supported such that the training experience is much the same as it is in a field environment.</p> <p><b>MODERATELY SUPPORTED (MODERATE):</b> Most of the TS/PM is supported. Sufficient cues are present within the simulation environment and appropriate responses are supported to permit practice of Command/Staff procedures and techniques; However, missing cues/responses will not cause negative training.</p> <p><b>NOT REQUIRED:</b> This TS/PM does not need to be supported in the simulation environment.</p> <p><i>* Indicates that comments are required.</i></p>	<p>Use the comment fields below to describe the Information Products and Delivery Means required to support the step/measure.</p> <p>Use the following format: <i>Product=Mean</i></p> <p>Example: Overlay=MCS,Estimate=Fax</p>
<p>* 1. MI company commander integrates MI company assets and capabilities into all plans and operations.</p>	<p><input type="radio"/> High  <input type="radio"/> Moderate*  <input checked="" type="radio"/> Not Required*  Comments:</p>	<p>Product=Mean</p>
<p>a. Task organizes and coordinates for space to employ ground surveillance radars to support the brigade reconnaissance and surveillance plan.</p>	<p><input type="radio"/> High  <input type="radio"/> Moderate*  <input checked="" type="radio"/> Not Required*  Comments:</p>	<p>Product=Mean</p>
<p>(1) Adjusts assets to ensure protection from friendly units.</p>	<p><input type="radio"/> High  <input type="radio"/> Moderate*  <input checked="" type="radio"/> Not Required*  Comments:</p>	<p>Product=Mean</p>
<p>(2) Disperses assets to sustain protection and applies operational security (OPSEC) measures.</p>	<p><input type="radio"/> High  <input type="radio"/> Moderate*  <input checked="" type="radio"/> Not Required*  Comments:</p>	<p>Product=Mean</p>

Figure 13. Evaluation of Training Audience Tasks.

Netpage: [Title] Training Audience Task: EMPLOY MILITARY INTELLIGENCE COMPANY ASSIST

URL: http://www.orden.usma.edu/henrycp/updates.asp

	*Comments: <input type="text"/>	
(1) Specific orders and requests (SOR).	<input type="radio"/> High <input type="radio"/> Moderate* <input checked="" type="radio"/> Not Required* *Comments: <input type="text"/>	Products=Means <input type="text"/>
(2) Collection plan.	<input type="radio"/> High <input type="radio"/> Moderate* <input checked="" type="radio"/> Not Required* *Comments: <input type="text"/>	Products=Means <input type="text"/>
(3) Intelligence synchronization matrix.	<input type="radio"/> High <input type="radio"/> Moderate* <input checked="" type="radio"/> Not Required* *Comments: <input type="text"/>	Products=Means Overlay=Map <input type="text"/>

Provide any additional information you feel necessary to conduct this task.

Task Level Test Comment:

Select Represented Units for this Task

☐ NONE  
☐ COMMAND GROUP AND STAFF, CORPS (100-15)  
☐ COMMAND GROUP AND STAFF, DIVISION (71-100)  
☐ COMMAND GROUP AND STAFF, MEDICAL (101-150)  
☐ COMMAND GROUP AND STAFF, INFANTRY BRIGADE (7-30)

**Figure 14. Evaluation of Training Audience Tasks (Continued).**

Once the *Submit Ratings* button is selected, three different Access® tables are written to store the data. The first stores the task level comments and the computed TPSC code and an example was previously provided in table 5. Table 6 depicts the second table that is updated which stores the individual task step/ performance measure level user opinions. This table is named in the database for its indexed name from the original linking table (Appendix A). The Visual Basic® Source Code listing for these functions is the same as that which provided the navigation screen for the training audience tasks (Appendix D). However, the screen would now be updated with the newly entered user entries.

**Table 6. Extract of Selection of Training Audience Tasks Database Table**

USERID	Session	Tasknumber	Stepnumber	PMNumber	Rating	ProdPipe	Comments
rude	24	1	2	3	high	Unit opnl status/situation reports=FM CMD Net	Actions of subordinate cdrs only
rude	24	1	2	2	high	situation reports=FM CMD Net	Routine situation reports from subordinate companies
rude	24	1	2	1	none		Cdr executes task
rude	24	1	1	5	none		Cdr executes task
rude	24	1	1	4	none		Staff provides data
rude	24	1	1	3	none		Cdr & Staff perform tasks
rude	24	1	1	2	none		Cdr & Staff conduct MA
rude	24	1	1	1	none		Staff provides input
rude	24	2	2	5	none		XO conducts task
rude	24	2	2	4	high	OPORD/FRAG O=BDE CMD Net	

Table 7 links the training audience task with selected represented units. The update column in this table is also used for navigation in the evaluation of represented units once all training audience tasks have been linked.

**Table 7. Extract of Linkage of Training Audience Task to Represented Unit**

USERID	Session	MTP	MTP_Task	RU	Update
mcnett	47	D0000077	3	D0000087	No
mcnett	47	D0000077	3	DMTP0002	No
mcnett	47	D0000077	3	D0000084	No
mcnett	47	D0000077	3	D0000081	No
mcnett	47	D0000077	46	D0000217	No
mcnett	47	D0000077	46	DMTP0003	Some
phelan	1317	D0000250	63	D0000083	No
phelan	1317	D0000250	63	D0000082	No

**REPRESENTED UNIT STATUS SCREEN:** Once all selected Training Audience Tasks have been evaluated, the T-RAP then allows a SME top move forward and begin to evaluate previously selected represented units for this training audience. Figure 15 depicts a sample screen that would be presented. The user is then asked to select first the

tasks that the represented unit would have to accomplish in order to support the training audience tasks. Once those tasks are selected, the user then evaluates each of the selected tasks. Figures 16 -18 show examples of the screens an user would use to evaluate represented units tasks.

WARSIM 2000 Subject Matter Expert Data Collection System  
Represented Unit Task Selection and Evaluation  
Training Audience - COMMAND GROUP AND STAFF, HEAVY BRIGADE (71-3)

Navigation:

Select Another Training Audience

Represented Units and Tasks to Be Evaluated

- Training Audience Task - EMPLOY MILITARY INTELLIGENCE COMPANY ASSETS 71-4-0018
  - First, Select Tasks to Evaluate for this Unit: [Redacted]
  - Then, Evaluate These Selected Tasks for this Unit: [Redacted]
- [Redacted]
  - First, Select Tasks to Evaluate for this Unit: [Redacted]
  - Then, Evaluate These Selected Tasks for this Unit: [Redacted]
- [Redacted]
  - First, Select Tasks to Evaluate for this Unit: [Redacted]
  - Then, Evaluate These Selected Tasks for this Unit: [Redacted]

Figure 15. Represented Unit Status Screen.

NetView - [Training Audience Task - EMPLOY MILITARY INTELLIGENCE COMPANY ASSETS 71-6-0018]

http://www.dcm.usna.edu/warcp/mapii.asp

Select Represented Unit Tasks for COMMAND GROUP AND STAFF, HEAVY BRIGADE (71-3)

Supporting Training Audience - COMMAND GROUP AND STAFF, HEAVY BRIGADE (71-3)

Conducting Training Audience Task - EMPLOY MILITARY INTELLIGENCE COMPANY ASSETS 71-6-0018

Select all of the Represented Unit tasks to support the Training Audience and its task in WARSIM:  
When complete, push the button at the bottom of this frame.

Task Number	Task Conditions Standards	Represented Unit TASK?
	ELEMENT: ELEMENT: BRIGADE MI	
71-6-0018	TASK: EMPLOY MILITARY INTELLIGENCE COMPANY ASSETS	<input checked="" type="checkbox"/>
	ELEMENT: ELEMENT: COORDINATING STAFF	
71-6-0019	TASK: PLAN RECONNAISSANCE AND SURVEILLANCE	<input type="checkbox"/>
	ELEMENT: ELEMENT: BRIGADE S2 SECTION	
71-6-0077	TASK: PROVIDE INTELLIGENCE INPUT TO THE COMMAND ESTIMATE	<input type="checkbox"/>
	ELEMENT: ELEMENT: HHC COMMANDANT	
71-6-0246	TASK: PLAN AND SUPERVISE MAIN COMMAND POST SECURITY	<input type="checkbox"/>
	ELEMENT: ELEMENT: BRIGADE S2 SECTION	
71-6-0811	TASK: CONDUCT INTELLIGENCE FUNCTIONS FOR DEPLOYMENT	<input type="checkbox"/>
	ELEMENT: ELEMENT: BRIGADE TOC, TAC, REAR CP	
71-6-1080	TASK: EMPLOY OPERATIONS SECURITY MEASURES	<input type="checkbox"/>
	ELEMENT: ELEMENT: BRIGADE PROVOST MARSHAL	
71-6-1208	TASK: PLAN SECURITY OF DESIGNATED PERSONS, UNITS, CONVOYS, FACILITIES, AND MSR CRITICAL POINTS	<input type="checkbox"/>
	ELEMENT: ELEMENT: BRIGADE S2 SECTION	
71-6-2001	TASK: PREPARE THE INTELLIGENCE ESTIMATE	<input type="checkbox"/>
71-6-2002	TASK: PREPARE THE INTELLIGENCE ANNEX TO THE OPERATION ORDER	<input type="checkbox"/>
71-6-2003	TASK: ANALYZE INCOMING INFORMATION FROM MANEUVER ELEMENTS IN CONJUNCTION WITH INTELLIGENCE RECEIVED FROM HIGHER HEADQUARTERS G2	<input type="checkbox"/>
71-6-2004	TASK: MANAGE THE INTELLIGENCE EFFORT	<input type="checkbox"/>
71-6-2004	TASK: DEPRESS SUBJECT INFORMATION DECONTAMINATE DATA	<input type="checkbox"/>

Figure 16. Selection of Represented Unit Tasks Screen.

Network: Training Audience Task - EMPLOY MILITARY INTELLIGENCE COMPANY ASSETS (71-6-0018)

http://www.ccsn.usma.edu/warpg/mapi2.asp

Support Home Profile Editor Publications Real Search Reports Summary

**Evaluate Represented Unit Tasks for COMMAND GROUP AND STAFF, HEAVY BRIGADE (71-3)**  
**Supporting Training Audience - COMMAND GROUP AND STAFF, HEAVY BRIGADE (71-3)**  
**Conducting Training Audience Task - EMPLOY MILITARY INTELLIGENCE COMPANY ASSETS 71-6-0018**

Evaluate all of the selected Represented Unit tasks to support the Training Audience and its task in WARSIM:

Task Number	Task Conditions Standards	Previously Evaluated?	Evaluate RU TASK
	ELEMENT: ELEMENT: BRIGADE MI		
71-6-0018	TASK: EMPLOY MILITARY INTELLIGENCE COMPANY ASSETS	No	
	ELEMENT: ELEMENT: COORDINATING STAFF		
71-6-0019	TASK: PLAN RECONNAISSANCE AND SURVEILLANCE	No	
	ELEMENT: ELEMENT: BRIGADE S2 SECTION		
71-6-0077	TASK: PROVIDE INTELLIGENCE INPUT TO THE COMMAND ESTIMATE	No	

Figure 17. Selection of Represented Unit Tasks Screen

Rate Steps and Performance Measures

Represented Unit - COMMAND GROUP AND STAFF, HEAVY BRIGADE (71-3)  
Performing Task EMPLOY MILITARY INTELLIGENCE COMPANY ASSETS  
in support of COMMAND GROUP AND STAFF, HEAVY BRIGADE (71-3)  
conducting task EMPLOY MILITARY INTELLIGENCE COMPANY ASSETS 71-6-0018

<p><b>TASK NUMBER:</b> 71-6-0018  <b>ELEMENT:</b> ELEMENT: BRIGADE MI  <b>TASK:</b> EMPLOY MILITARY INTELLIGENCE COMPANY ASSETS (71-6-0018 KBM3)  <b>CONDITIONS:</b> The brigade is preparing for or is engaged in combat. The brigade commander has designated the current task organization. The S2 is located at the main command post (CP). The main CP is operational. The S3 is developing or has issued an operation order (OPORD) or operation plan (OPLAN). Some iterations of this task should be performed in MOPPA4.  <b>STANDARDS:</b> The military intelligence (MI) company assets are fully used to support operations. Time required to plan and implement is increased when performing in MOPPA4.</p>	<p><b>TSSC TASK STEP PERFORMANCE MEASURE RATINGS</b></p> <p>Given this Represented Unit performs this task in support of your Training Audience task, to what extent do you require the details of the accomplishment of this step/ performance measure?</p> <p><b>FULL DETAILS:</b> All aspects of the RU's TS/ PM must be modeled in the simulation to stimulate the associated Training Audience Task.</p> <p><b>PARTIAL DETAILS:</b> Only essential aspects of this RU's TS/ PM must be modeled in the simulation to stimulate the associated Training Audience Task.</p> <p><b>MINIMUM DETAILS:</b> All this is required is to know the RU TS/ PM has been executed (Yes/No).</p> <p><b>NOT REQUIRED:</b> No Information Required.</p> <p><i>* Indicates that comments are required.</i></p>	<p>Enter Represented Unit Major End Items of Equipment and Simulation Synthetic Environment Objects.</p> <p>Use the following format:  item, item</p> <p>Example: bridge, river</p>
<p>* 1. MI company commander integrates MI company assets and capabilities into all plans and operations.</p>	<p>C Full Details  C Partial Details  C Minimum Details  @ Not Required  * Comments:</p>	<p>Equipment:  <input type="text"/>  SE Objects:  <input type="text"/></p>
<p>* 4. Task organizes and coordinates for space to employ ground surveillance radars to support the brigade reconnaissance and surveillance plan.</p>		

Figure 18. Evaluation of Represented Unit Tasks Screen

In conjunction with the processing of these screens, several tables are updated within the Access® databases to reflect the user's responses. Tables 8 provides an example of the linkages created between training audience tasks and represented unit tasks. Table 9 shows the data collected at the represented unit task level. Note here that now the system is asking the SME to provide input on any synthetic objects and equipment that the simulation must provide the simulated unit in order for it to adequately perform the indicated task. Appendices F through H provide the source code for the selection and evaluation of represented unit tasks and dynamically produce the

source code for Figures 15-18. The methodology used to populate these tables is the same as that for the training audience data.

**Table 8. Extract of Linkage of Training Audience Task to Represented Unit Task**

USERID	MTP	MTP_Task	RU	RU_Task	Update	TSSC
mcnett	D0000077	3	DMTP0002	14	No	0
mcnett	D0000077	3	DMTP0002	13	No	0
mcnett	D0000077	3	DMTP0002	12	No	0
mcnett	D0000077	46	DMTP0003	72	No	0
mcnett	D0000077	46	DMTP0003	26	No	0
mcnett	D0000077	46	DMTP0003	23	No	0
phelan	D0000250	38	DMTP0003	2	Yes	0
phelan	D0000250	38	DMTP0003	1	Yes	0
phelan	D0000250	38	DMTP0003	84	Yes	0
phelan	D0000250	38	DMTP0003	83	Yes	0

**Table 9. Extract of Represented Unit Task Detailed Data**

USERID	MTP	MTP_Task	RU_Task	RU_Step	RU_PM	Rating	Equip	Object	Comment
hughesj	D0000239	35	1	6	2	none			
hughesj	D0000239	35	1	6	1	low			That this is done or not.
Hughesj	D0000239	35	1	5	5	med			The early warning needs to be reported to the PTA BN TOC/CP. The report needs to be the standard early warning and air detection report.
hughesj	D0000239	35	1	5	4	low	EWBN System		That this is done or not.
hughesj	D0000239	35	1	5	3	low	FLIR System		That this is done or not.
hughesj	D0000239	35	1	5	2	low			That this is done or not.

## 7.0 OUTPUT ANALYSIS

One of the many benefits of the design of the T-RAP is that once data begins to populate the database, it is relatively easy using Standard Query Language (SQL) queries to analyze what has been entered. This section describes the design of the query engine which an user can run from the initial login page. It also provides a description of the various query products designed by the T-RAP team to support the WARSIM 2000 requirements process

As the Access® databases became populated with user opinions, the T-RAP developers began to develop a series of queries to extract the entered information and present it to authorized users using the web as the transmission medium. Two Visual Basic® CGI programs were written to create this query engine. The first, dynamically creates an option page for the user. Once the query engine is selected, the program extracts from the database, a listing of all current users, training audiences and represented units. These three lists are presented in HTML multiple select boxes. Figure 19 provides a screen capture of what the user is presented. The intent is for an user to be able to quickly select a subset of any combination of these three options (user, training audience, represented unit) before selecting any query. Appendix J provides the Visual Basic® source code for the dynamic construction of this screen.

WARSIM 2000 Query Engine Input Form

This query form will present data as it currently resides in the WARSIM 2000 database. Select how you want to look at the data. You may look at all or any combination of users, Training Audiences and Represented Units. Feel free to experiment until you find what you are looking for. More queries are in the design stage now. I am also looking for any comments as to other queries that may be useful.

*MAJ Rob Phillips*

Select Which User(s) Data you want included in the Query

- COL James Shaffer
- MAJ George Stone
- MAJ Frank Rhinesmith
- Dir. ATISD Carol Huber
- Mr. Alan R. Moeller
- Analyst Gerald M. Post
- DADCST Frank Polster
- CPT Stephen Schless
- CPT Robert C. Logsdon

Select Which Training Audience(s) you want specifically included in the Query

- COMMAND GROUP AND STAFF, CORPS (100-15)
- COMMAND GROUP AND STAFF, CORPS (100-15) (Pending)
- COMMAND GROUP AND STAFF, DIVISION (71-100)
- COMMAND GROUP AND STAFF, HEAVY BRIGADE (71-3)
- COMMAND GROUP AND STAFF, INFANTRY BRIGADE (7-30)
- ADA: GUN OR STRINGER BATTALION (44-115)
- ADA: STINGER PLATOON (44-117-11)
- ADA: VULCAN PLATOON (TOWED) (44-117-12)
- ADA: AVENGER PLATOON (44-117-21)

Select Which Represented Unit(s) you want specifically included in the Query

- COMMAND GROUP AND STAFF, CORPS (100-15)
- COMMAND GROUP AND STAFF, CORPS (100-15) (Pending)
- COMMAND GROUP AND STAFF, DIVISION (71-100)
- COMMAND GROUP AND STAFF, HEAVY BRIGADE (71-3)
- COMMAND GROUP AND STAFF, INFANTRY BRIGADE (7-30)
- ADA: GUN OR STRINGER BATTALION (44-115)
- ADA: STINGER PLATOON (44-117-11)
- ADA: VULCAN PLATOON (TOWED) (44-117-12)

**Figure 19. Standard Query Screen**

The lower portion of the query page presents the user with approximately 20 standard queries. These queries represent what STRICOM/ NSC as well as the developers felt were important to support the data analysis. Each query is listed and a brief description provided. The queries were designed in Access® using its graphical interface and the resulting SQL code was cut and pasted into the source code which is provided in Appendix K.

- SME Work Status - Provides a management tool which allows users to see progress by individual users.

- TPSCs by SME - Individual - Query provides option to look at computed TPSC data. With individual comments
- TPSCs by SME - All - Query provides the user with the ability to combine and compare computed TPSC s
- TA Task to Products and Pipes - This query provides a detailed cross reference of all tasks to user entered comments on communications links and products.
- TPSCs ordered by TPSC
- TA Task Companion List - Query requested Veda, INC to identify an task step/ performance measure that was given a rating of 'High'
- TA Task Exclusion List - Another query requested by Veda to identify those tasks not required in the requirements process.
- RU frequency by TA Task - A cross reference query to identify how many times a particular represented unit task was selected to support a training audience task
- RUs by TA Task - Another cross reference query identifying represented units to training audience tasks.
- TA frequency by RU Task - This query provides the ability to see how often a training audience is supported by a particular represented unit task
- TAs by RU Task - Similar query but references to the training audience level only.
- TSSCs by SME - TA Task to RU Task - Similar to TPSC query but with the added dimension of the relationship of training audience task to represented unit task
- TSSCs by SME - RU Task to TA Task - Similar to previous but referencing represented unit task to training audience task.

- RU Task frequency by TA - Provides a count of how many times a represented unit task was selected to support a particular training audience
- TSSCs - All SMEs - Ability to see TSSCs rolled up.
- TSSCs All SMEs - TA Task to RU Task - Combination of queries to see computed TSSC code by user related to training audience task.
- TSSCs All SMEs - RU Task to TA Task - same as previous but indexed on the training audience task
- RU Exclusion List - Lists those tasks not necessary within the simulation
- RU Companion List - Identifies any represented unit task step/ performance measure that was given a TSSC SME rating of high.

Each of these queries also generate a text based version of the output that an user can download and input into Excel® or Access® for further analysis.

## 8.0 LESSONS LEARNED

As the T-RAP was developed, it became apparent that the Access® database engine was not robust enough to meet user performance expectations. Also, using CGI placed most of the processing burden on the server with little or no work accomplished by the user platforms. Also, STRICOM realized that the true worth of the T-RAP would be in its ability to merge multiple expert opinions into one consensus for the Army. The developed a combining algorithm for the multiple opinions but it was never implemented in this version of the tool. Their algorithm is included in Appendix L and will be included in the next version.

## 9.0 CONCLUSIONS AND FUTURE WORK

The initial prototype of T-RAP was quickly developed because of the use of Microsoft® Access® and Visual Basic®. Because there was a desire to complete the initial T-RAP quickly, decisions on implementation methods were made to help us meet the deadlines. The first version of T-RAP was based on the Microsoft® Jet database as the backend. T-RAP had several different databases, each fulfilling different data requirements within T-RAP. The frontend was entirely based on Common Gateway Interfaces (CGI) using Microsoft® Visual Basic® as the development language. All three of these implementation choices led to a rapid development of the tool, but also led to efficiency problems on the server and across the network.

First, the Jet Database engine did not support high levels of concurrency, thus slowing the access to the database at times. Next, four different databases, with multiple tables in each database, forced the CGI applications to open and close several of the databases multiple times within one operation within T-RAP. Lastly, all of the processing was accomplished on the T-RAP server through CGI. The CGI would receive a request from a browser, process the request through the jet database engine, dynamically build the pages, and distribute each "new" page back to the browser. As a result, much of the information being passed to the browser from the web server was formatting information such as HTML formatting tags. This alone created a vast amount of overhead on the network as the page formatting tags started taking up larger percentages of the network traffic than the raw data.

Since June 1997 we have been developing the next iteration of T-RAP. This version has fixed most of the disadvantages of the CGI / jet database T-RAP. This latest

version will be created using Powersoft Corporation's Powerbuilder 5.0 frontend, and SYBASE SQL Server as the backend. With these changes, only raw data and SQL statements are being sent across the network. The actual formatting of this data is handled by the browser using a Powerbuilder plugin. By doing so, the database remains consistent, and users are much happier since the overall application is much more efficient. In addition to making the overall application more efficient and user friendly, we also incorporated additional information in the database to make it easier to modify in the future.

The T-RAP has been a success for both the ORCEN, D/EECS and STRICOM/NSC. It demonstrated an advanced concept in the requirements documentation process and in our ability to use tools such as this in the future.

## 10.0 REFERENCES

- Gundavaram, S. (1996). CGI Programming on the World Wide Web. O'Reilly & Associates, INC.
- Lockeed Martin, INC. (1996) WARSIM 2000 System Specification, Orlando, FL, STRICOM
- Sherikon, INC. (1995). Task Performance Support Codes,, Technical Report, US Army Simulation Training and Instrumentation Command, Orlando, FL, STRICOM.
- US Army STRICOM, (1997), Operational Requirements Document for Warfighters' Simulation (WARSIM) 2000., Orlando FL, STRICOM.

# APPENDIX A - SUPPORTED MISSION TRAINING PLANS

FILENAME	DOCID	DOC DATE	DOC TITLE
D0000001.dbf	100-15-MTP	5/1/90	CORPS COMMAND GROUP AND STAFF
D0000999.dbf	100-15-MTP(Pending)	4/1/97	CORPS COMMAND GROUP AND STAFF (Pending)
D0000210.dbf	71-100-MTP	5/1/90	DIVISION COMMAND GROUP AND STAFF
D0000250.dbf	71-3-MTP	10/3/88	HEAVY BRIGADE COMMAND GROUP AND STAFF
D0000240.dbf	7-30-MTP	2/17/89	INFANTRY BRIGADE (COMMAND GROUP/STAFF)
D0000192.dbf	44-115-MTP	2/10/92	ADA BATTALION, GUN OR STRINGER
D0000193.dbf	44-117-11-MTP	2/5/92	ADA PLATOON, STINGER
D0000194.dbf	44-117-12-MTP	7/13/92	PLATOON, VULCAN (TOWED)
D0000185.dbf	44-117-21-MTP	6/10/92	AVENGER PLATOON
D0000195.dbf	44-117-30-MTP	6/5/92	ADA BATTERY, GUN OR STINGER
D0000196.dbf	44-117-31-MTP	6/15/92	AVENGER BATTERY
D0000186.dbf	44-445-MTP	4/28/92	ADA BATTALION, CHAPARRAL
D0000187.dbf	44-447-10-MTP	4/20/92	ADA PLATOON, CHAPARRAL
D0000188.dbf	44-447-30-MTP	6/18/92	ADA BATTERY, CHAPARRAL
D0000189.dbf	44-495-MTP	9/3/91	AN ADA BATTALION, HAWK
D0000190.dbf	44-497-30-MTP	11/20/92	ADA BATTERY, HAWK
D0000197.dbf	44-635-MTP	9/2/91	ADA BATTALION, PATRIOT
D0000191.dbf	44-637-30-MTP	9/2/91	ADA BATTERY, PATRIOT
D0000175.dbf	12-402-MTP	9/30/93	HEADQUARTERS AND HEADQUARTERS DETACHMENT, COMMAND GROUP AND STAFF, PERSONNEL GROUP
D0000178.dbf	12-426-MTP	5/7/93	PERSONNEL SERVICE BATTALION
D0000184.dbf	77-004-10-MTP	4/1/88	G1-AG SECTION
D0000198.dbf	12-113-MTP	7/14/87	DIVISION AND ARMY BAND
D0000018.dbf	12-606-60-MTP	9/30/93	HEADQUARTERS AND HEADQUARTERS DETACHMENT, COMMAND GROUP AND STAFF, REPLACEMENT COMPANY/BATTALION
D0000179.dbf	12-606-61-MTP	9/19/90	HEADQUARTERS AND HEADQUARTERS DETACHMENT, COMMAND GROUP AND STAFF, REPLACEMENT BATTALION/COMPANY (CRC)
D0000177.dbf	12-413-30-MTP	9/30/93	POSTAL COMPANY (DIRECT SUPPORT/GENERAL SUPPORT)
D0000247.dbf	17-442-MTP	9/30/93	ARMORED CALVARY REGIMENT COMMAND GROUP
D0000238.dbf	71-2-MTP	10/3/88	TANK AND MECHANIZED INFANTRY BATTALION TASK FORCE
D0000248.dbf	17-485-MTP	5/7/93	REGIMENTAL ARMORED CAVALRY SQUADRON
D0000246.dbf	17-385-MTP	6/30/93	DIVISION CAVALRY SQUADRON
D0000235.dbf	71-1-MTP	10/3/88	TANK AND MECHANIZED INFANTRY COMPANY AND COMPANY TEAM
D0000249.dbf	17-487-30-MTP	9/3/91	REGIMENTAL ARMORED CAVALRY TROOP
DMTP0009.dbf	17-237-10-MTP	7/9/96	Tank Platoon
D0000230.dbf	17-57-10-MTP	12/27/88	SCOUT PLATOON
DMTP0008.dbf	17-57-10-MTP	1/1/95	SCOUT PLATOON

D0000228.dbf	17-236-10-MTP	12/2/87	TASK FORCE MAINTENANCE PLATOON
D0000229.dbf	17-236-11-MTP	11/25/87	TASK FORCE SUPPORT PLATOON
D0000244.dbf	17-236-12-MTP	12/2/87	TASK FORCE MEDICAL PLATOON
D0000081.dbf	1-100-MTP	6/7/90	AVIATION BRIGADE AND BATTALION
D0000082.dbf	1-100-30-MTP	12/6/91	HEADQUARTERS AND HEADQUARTERS COMPANY AVIATION BRIGADE AND BATTALION
DMTP0003.dbf	1-045-MTP	10/3/95	Assault Aviation Battalion
DMTP0005.dbf	1-215-MTP	1/29/96	Command Aviation Battalion
DMTP0001.dbf	1-385-MTP	10/3/95	Attack Helicopter Battalion
DMTP0004.dbf	1-425-MTP	4/5/96	Air Traffic Control Battalion
D0000224.dbf	1-947-30-MTP	6/21/91	AVIATION INTERMEDIATE MAINTENANCE (AVIM) COMPANY/BATTALION, CORPS SUPPORT COMMAND (COSCOM), AND THEATER AREA ARMY COMD (TAACOM)
D0000222.dbf	1-946-MTP	1/8/92	BN HQ, AVIATION INTERMEDIATE MAINTENANCE (AVIM) BATTALION, CORPS SUPPORT COMMAND (COSCOM); AND THEATER ARMY AREA COMMAND (TAACOM)
D0000085.dbf	1-187-30-MTP	5/18/89	ATTACK HELICOPTER COMPANY
D0000083.dbf	1-103-30-MTP	8/18/89	ASSAULT HELICOPTER COMPANY
D0000088.dbf	1-247-30-MTP	10/19/89	MEDIUM HELICOPTER COMPANY
D0000084.dbf	1-108-30-MTP	5/30/89	AIR CAVALRY/RECONNAISSANCE TROOP
DMTP0002.dbf	1-167-30-MTP	10/3/95	Air Reconnaissance Troop
D0000086.dbf	1-217-30-MTP	9/30/93	COMMAND/THEATER AVIATION COMPANY
D0000217.dbf	1-129-30-MTP	1/8/90	AVN UNIT MAINT (AVUM) CO COMBAT AVN BN AND THE AVN UNIT MAINT (AVUM) TROOP REGIMENTAL COMBAT AVN SQUADRON
D0000223.dbf	1-946-30-MTP	1/30/92	HEADQUARTERS AND HEADQUARTERS DETACHMENT, AVIATION INTERMEDIATE MAINTENANCE (AVIM) BATTALION, COSCOM/TAACOM
D0000218.dbf	1-926-MTP	6/20/91	BATTALION HEADQUARTERS, AVIATION MAINTENANCE BATTALION, AIR ASSAULT DIVISION
D0000219.dbf	1-926-30-MTP	3/29/90	HEADQUARTERS, HEADQUARTERS AND SUPPLY COMPANY, AVIATION MAINTENANCE BATTALION AIR ASSAULT DIVISION
D0000220.dbf	1-927-30-MTP	9/5/90	AVIATION INTERMEDIATE MAINTENANCE (AVIM) COMPANY AVIATION MAINTENANCE BATTALION AIR ASSAULT DIVISION
D0000221.dbf	1-933-30-MTP	9/13/89	AVIATION INTERMEDIATE MAINTENANCE (AVIM) COMPANY, DIVISION SUPPORT COMMAND (DISCOM)
D0000087.dbf	1-227-10-MTP	4/16/88	AIR TRAFFIC CONTROL PLATOON
D0000089.dbf	1-427-10-MTP	3/8/94	AIR TRAFFIC SERVICES EN ROUTE CONTROL PLATOON
D0000068.dbf	41-705-MTP	3/22/92	CIVIL AFFAIRS BATTALION (GP)
D0000069.dbf	41-707-30-MTP	3/19/93	DETACHMENT (GENERAL SUPPORT), CIVIL AFFAIRS BATTALION (GP)
D0000070.dbf	41-715-MTP	4/20/92	CIVIL AFFAIRS BATTALION (FID/UW)
D0000067.dbf	41-701-30-MTP	3/22/92	HEADQUARTERS AND HEADQUARTERS COMPANY, CIVIL AFFAIRS COMMAND
D0000037.dbf	5-145-MTP	9/30/92	HEADQUARTERS AND HEADQUARTERS COMPANY, ENGINEER BATTALION, HEAVY DIVISION/CORPS
D0000039.dbf	5-415-MTP	9/30/91	HEADQUARTERS AND SUPPORT COMPANY, ENGINEER BATTALION (COMBAT HEAVY)
D0000011.dbf	5-025-MTP	9/30/92	HHC, ENGINEER SUPPORT BATTALION
D0000017.dbf	5-605-MTP	6/15/92	HEADQUARTERS AND HEADQUARTERS COMPANY

D0000013.dbf	5-145-31-MTP	2/17/89	ENGINEER COMPANY, HEAVY DIVISION/CORPS/ARMORED CAVALRY REGIMENT
D0000036.dbf	5-025-31-MTP	10/18/89	COMBAT ENGINEER COMPANY, (AIRBORNE/LIGHT/AIR ASSAULT/MOTORIZED DIVISION AND AIRBORNE CORPS
D0000016.dbf	5-415-31-MTP	9/30/91	ENGINEER COMPANY, ENGINEER BATTALION (COMBAT HEAVY)
D0000038.dbf	5-145-32-MTP	7/19/91	ENGINEER BRIDGE COMPANY
D0000232.dbf	5-145-11-MTP	2/1/89	COMBAT ENGINEER PLATOON, HEAVY DIVISION/CORPS/ARMORED CAVALRY REGIMENT
D0000231.dbf	5-025-11-MTP	10/18/89	COMBAT ENGINEER PLATOON, (AIRBORNE/LIGHT/AIR ASSAULT/MOTORIZED DIVISION AND AIRBORNE CORPS)
D0000012.dbf	5-145-12-MTP	7/19/91	ENGINEER BRIDGE PLATOON
D0000014.dbf	5-415-13-MTP	9/30/91	ENGINEER EQUIPMENT PLATOONS
D0000233.dbf	5-415-14-MTP	9/30/91	ENGINEER MAINTENANCE PLATOONS
D0000015.dbf	5-415-17-MTP	9/30/91	ENGINEER CONSTRUCTION PLATOONS
D0000040.dbf	5-423-11-MTP	9/30/91	ENGINEER DUMP TRUCK PLATOON
D0000234.dbf	5-605-11-MTP	6/15/92	ENGINEER TOPOGRAPHIC PLATOONS/DETACHMENTS
D0000253.dbf	6-115-20-MTP	1/24/90	FIELD ARTILLERY CANNON BATTALION FIRE SUPPORT
D0000252.dbf	6-115-MTP	11/23/90	FA CANNON BATTALION HEADQUARTERS AND HEADQUARTERS BATTERY; HEADQUARTERS, HEADQUARTERS AND SERVICE BATTERY; OR SERVICE BATTERY
D0000257.dbf	6-525-MTP	1/11/90	FIELD ARTILLERY MULTIPLE LAUNCH ROCKET SYSTEM BATTALION
D0000251.dbf	6-037-30-MTP	3/28/89	CANNON FIRING BATTERY 155-MM
D0000170.dbf	6-127-30-MTP	1/24/90	CANNON FIRING BATTERY 105-MM, TOWED
D0000254.dbf	6-167-30-MTP	1/24/90	CANNON FIRING BATTERY (3 x 6), 155-MM, TOWED, M198/M114
D0000173.dbf	6-397-30-MTP	11/20/90	CANNON FIRING BATTERY 8-INCH, SELF-PROPELLED
D0000172.dbf	6-367-30-MTP	11/20/90	FIELD ARTILLERY CANNON BATTERY (3X8) BATTERY ADMINISTRATION AND LOGISTICS
D0000174.dbf	6-398-30-MTP	7/8/89	FIELD ARTILLERY MULTIPLE LAUNCH ROCKET SYSTEM BATTERY
D0000171.dbf	6-303-30-MTP	12/6/91	FA TARGET ACQUISITION BATTERY AND THE CORPS TARGET ACQUISITION DETACHMENT
D0000255.dbf	6-367-20-MTP	12/20/90	FIELD ARTILLERY CANNON BATTERY FIRING PLATOON 155-MM, SELF PROPELLED AND TOWED
D0000256.dbf	6-447-20-MTP	12/20/90	FA CANNON BATTERY FIRING PLATOON 8-INCH, SELF PROPELLED
D0000183.dbf	14-612-MTP	9/25/90	THEATER FINANCE COMMAND
D0000180.dbf	14-412-MTP	7/3/89	FINANCE GROUP (FG)
D0000182.dbf	14-426-MTP	5/7/93	FINANCE BATTALION
D0000181.dbf	14-423-30-MTP	9/29/94	FINANCE DETACHMENT
D0000239.dbf	7-20-MTP	12/27/88	INFANTRY BATTALION
D0000227.dbf	7-94-MTP	10/12/89	INFANTRY BATTALION HEADQUARTERS AND HEADQUARTERS COMPANY AND COMBAT SUPPORT/COMBAT SERVICE SUPPORT PLATOONS
D0000237.dbf	7-10-MTP	10/3/88	INFANTRY RIFLE COMPANY
D0000243.dbf	7-91-MTP	10/12/88	ANTIARMOR COMPANY/PLATOON/SECTION
D0000226.dbf	7-93-MTP	12/27/88	LONG-RANGE SURVEILLANCE COMPANY/DETACHMENT TEAM (HVY/LT)
D0000241.dbf	7-8-MTP	9/30/88	INFANTRY RIFLE PLATOON AND SQUAD
DMTP0006.dbf	7-8-94-MTP	1/1/95	Infantry Rifle Platoon and Squad

D0000242.dbf	7-90-MTP	1/8/89	INFANTRY MORTAR PLATOON, SECTION, AND SQUAD
D0000225.dbf	7-92-MTP	3/16/89	INFANTRY SCOUT PLATOON/SQUAD AND SNIPER TEAM
D0000114.dbf	8-611-MTP	10/1/93	HEADQUARTERS, MEDICAL COMMAND
D0000135.dbf	8-422-MTP	3/20/92	HEADQUARTERS, MEDICAL BRIGADE
D0000213.dbf	8-422-30-MTP	11/17/93	HEADQUARTERS COMPANY
D0000136.dbf	8-432-MTP	11/17/93	HEADQUARTERS MEDICAL GROUP
D0000214.dbf	8-456-MTP	9/30/93	HEADQUARTERS AREA SUPPORT MEDICAL BATTALION
D0000143.dbf	8-446-MTP	9/20/94	MEDICAL EVACUATION BATTALION HEADQUARTERS
D0000144.dbf	8-446-30-MTP	11/19/93	HEADQUARTERS DETACHMENT, MEDICAL EVACUATION BATTALION
D0000139.dbf	8-475-MTP	9/30/91	MEDICAL BATTALION HEADQUARTERS (DENTAL SERVICE)
D0000215.dbf	8-485-MTP	9/30/93	HEADQUARTERS MEDICAL BATTALION, LOGISTICS (FORWARD AND REAR)
D0000149.dbf	8-XXX-30-MTP	4/30/94	COMMON COMPANY TASKS - MEDICAL
D0000211.dbf	8-028-30-MTP	8/25/92	AEROMEDICAL EVACUATION UNITS
D0000112.dbf	8-409-MTP	6/4/93	MEDICAL DETACHMENT, VETERINARY SERVICE HEADQUARTERS
D0000134.dbf	8-057-30-MTP	3/18/92	MEDICAL COMPANY MAIN SUPPORT BATTALION HEAVY DIVISION
D0000212.dbf	8-058-30-MTP	9/24/93	MEDICAL COMPANY, FORWARD SUPPORT BATTALION, SUPPORT COMMAND, HEAVY DIVISION
D0000142.dbf	8-417-30-MTP	11/1/93	MEDICAL DETACHMENT
D0000092.dbf	8-418-30-MTP	11/1/93	MEDICAL DETACHMENT, VETERINARY MEDICINE
D0000093.dbf	8-437-30-MTP	9/30/93	MEDICAL CO, SUPPORT BN, HEAVY SEPARATE BRIGADE/SEPARATE INFANTRY BRIGADE, AND MEDICAL TROOP, SUPPORT SQUADRON, ARMORED CAVALRY REG
D0000094.dbf	8-449-30-MTP	2/25/92	MEDICAL COMPANY GROUND AMBULANCE
D0000113.dbf	8-456-30-MTP	11/1/93	SUPPORT COMPANY, AREA SUPPORT MEDICAL BATTALION
D0000137.dbf	8-457-30-MTP	4/14/93	AREA SUPPORT MEDICAL COMPANY (ASMC)
D0000145.dbf	8-458-30-MTP	6/4/93	MEDICAL COMPANY (HOLDING)
D0000138.dbf	8-467-30-MTP	11/30/90	MEDICAL COMPANY OR DETACHMENT COMBAT STRESS CONTROL
D0000095.dbf	8-476-30-MTP	5/22/91	HEADQUARTERS AND HEADQUARTERS DETACHMENT (DENTAL SERVICE), MEDICAL BATTALION (DENTAL SERVICE)
D0000146.dbf	8-478-30-MTP	4/30/91	MEDICAL COMPANY/DETACHMENT (DENTAL SERVICE)
D0000147.dbf	8-487-30-MTP	5/15/91	LOGISTICS SUPPORT COMPANY/DETACHMENT AND DISTRIBUTION COMPANY (FORWARD) AND (REAR)
D0000140.dbf	8-498-30-MTP	4/14/93	MEDICAL DETACHMENT, PREVENTIVE MEDICINE (ENTOMOLOGY) AND MEDICAL DETACHMENT, PREVENTIVE MEDICINE (SANITATION)
D0000148.dbf	8-667-30-MTP	3/10/93	AREA MEDICAL LABORATORY
D0000096.dbf	8-705-MTP	9/30/91	COMBAT SUPPORT HOSPITAL
D0000097.dbf	8-715-MTP	11/19/93	FIELD HOSPITAL
D0000216.dbf	8-725-MTP	5/7/93	GENERAL HOSPITAL
D0000141.dbf	8-765-30-MTP	12/9/91	MOBILE ARMY SURGICAL HOSPITAL (MASH)
D0000115.dbf	8-897-MTP	10/1/93	MEDICAL CENTER (THEATER MATERIEL MANAGEMENT)
D0000045.dbf	34-196-30-MTP	6/16/89	HEADQUARTERS COMPANY MILITARY INTELLIGENCE BRIGADE OR BATTALION
D0000046.dbf	34-268-15-MTP	9/30/91	GROUND SURVEILLANCE PLATOON INTELLIGENCE AND SURVEILLANCE COMPANY
D0000047.dbf	34-296-10-MTP	12/20/90	MILITARY INTELLIGENCE BATTALION
			MILITARY INTELLIGENCE COMBAT SERVICE SUPPORT UNITS

D0000044.dbf	34-10-20-1/2	12/14/92	MILITARY INTELLIGENCE COLLECTIVE TRAINING STANDARDS DOCUMENT VOLUME I & II
D0000108.dbf	19-472-MTP	9/19/90	HEADQUARTERS AND HEADQUARTERS COMPANY, MILITARY POLICE BRIGADE CORPS OR TAACOM
D0000109.dbf	19-476-MTP	3/29/90	HEADQUARTERS AND HEADQUARTERS DETACHMENT, MILITARY POLICE BATTALION
D0000106.dbf	19-17-30-MTP	10/3/88	DIVISION MILITARY POLICE COMPANY HEADQUARTERS/PROVOST MARSHAL STAFF
D0000111.dbf	19-77-30-MTP	2/17/89	MILITARY POLICE COMPANY (COMBAT SUPPORT)
D0000107.dbf	19-283-30-MTP	6/30/92	MILITARY POLICE DETACHMENT (CID) DIVISION SUPPORT ELEMENT CORPS/THEATER/PORT AREA SUPPORT ELEMENTS
D0000110.dbf	19-77-10-MTP	9/30/87	CORPS MILITARY POLICE PLATOON
D0000125.dbf	3-116-MTP	10/23/90	CHEMICAL BRIGADE OR BATTALION
D0000128.dbf	3-257-30-MTP	12/27/88	Chemical Company Headquarters (ALL)
D0000132.dbf	3-7-10-MTP	1/30/90	MECHANIZED SMOKE PLATOON
D0000130.dbf	3-447-10-MTP	7/26/91	MOTORIZED SMOKE PLATOON
D0000133.dbf	3-7-11-MTP	12/27/88	FUEL SUPPLY PLATOON
D0000127.dbf	3-207-10-MTP	10/23/89	NBC RECONNAISSANCE PLATOON
D0000129.dbf	3-417-10-MTP	2/25/92	DECONTAMINATION PLATOON
D0000131.dbf	3-457-10-MTP	9/30/92	SMOKE/DECONTAMINATION PLATOON
D0000126.dbf	3-117-40-MTP	10/3/88	CHEMICAL SECTION AND NBC CENTER
D0000202.dbf	9-062-MTP	7/13/90	HEADQUARTERS, ORDNANCE GROUP, CONVENTIONAL AMMUNITION, DIRECT SUPPORT/GENERAL SUPPORT
D0000209.dbf	9-686-MTP	8/8/91	HEADQUARTERS, ORDNANCE BATTALION, AMMUNITION
D0000205.dbf	9-066-MTP	7/13/90	HEADQUARTERS, ORDNANCE BATTALION, CONVENTIONAL AMMUNITION, DIRECT SUPPORT/GENERAL SUPPORT
D0000207.dbf	9-527-MTP	9/8/93	ORDNANCE DETACHMENT, EXPLOSIVE ORDNANCE DISPOSAL CONTROL TEAM
D0000203.dbf	9-062-30-MTP	9/11/90	HEADQUARTERS COMPANY
D0000204.dbf	9-064-30-MTP	9/11/90	ORDNANCE COMPANY, CONVENTIONAL AMMUNITION, DIRECT SUPPORT/GENERAL SUPPORT
D0000201.dbf	9-008-30-MTP	2/6/90	ORDNANCE MISSILE SUPPORT COMPANY, MAIN SUPPORT BATTALION, HEAVY DIVISION
D0000206.dbf	9-483-30-MTP	6/15/92	ORDNANCE COMPANY, AMMUNITION, DIRECT SUPPORT, MANEUVER-ORIENTED AMMUNITION DISTRIBUTION SYSTEM
D0000008.dbf	43-436-MTP	11/16/90	BATTALION HEADQUARTERS, ORDNANCE (MAINTENANCE) BATTALION, CSG/TAACOM
D0000003.dbf	43-056-MTP	5/7/93	BATTALION HEADQUARTERS ORDNANCE (MAINTENANCE) BATTALION AIRBORNE/AIR ASSAULT DIVISION
D0000005.dbf	43-146-MTP	8/29/89	BATTALION HQ, HQ AND MAIN SUPPORT COMPANY, ORDNANCE (MAINTENANCE) BATTALION, SUPPORT COMMAND, LIGHT INFANTRY DIVISION
D0000002.dbf	43-007-30-MTP	6/8/89	LIGHT ORDNANCE (MAINTENANCE) COMPANY, MAIN SUPPORT BATTALION, HEAVY DIVISION
D0000019.dbf	43-008-30-MTP	6/8/89	HEAVY ORDNANCE (MAINTENANCE) COMPANY, MAIN SUPPORT BATTALION, HEAVY DIVISION

D0000020.dbf	43-009-30-MTP	6/8/89	ORDNANCE (MAINTENANCE) COMPANY, FORWARD SUPPORT BATTALION, HEAVY DIVISION
D0000021.dbf	43-056-30-MTP	11/7/89	HEADQUARTERS AND LIGHT ORDNANCE (MAINTENANCE) BATTALION AIRBORNE/AIR ASSAULT DIVISION
D0000022.dbf	43-057-30-MTP	11/7/89	FORWARD SUPPORT MAINTENANCE COMPANY, ORDNANCE (MAINTENANCE) BATTALION, AIRBORNE/AIR ASSAULT/LIGHT INFANTRY DIVISION
D0000004.dbf	43-058-30-MTP	2/13/90	HEAVY ORDNANCE (MAINTENANCE) COMPANY, ORDNANCE (MAINTENANCE) BATTALION, AIRBORNE/AIR ASSAULT DIVISION
D0000023.dbf	43-079-30-MTP	6/3/91	ORDNANCE (MAINTENANCE) COMPANY, SUPPORT BATTALION, SEPARATE INFANTRY BRIGADE AND HEAVY SEPARATE BRIGADE
D0000006.dbf	43-146-30-MTP	8/29/89	HEADQUARTERS AND MAIN SUPPORT COMPANY, ORDNANCE (MAINTENANCE) BATTALION, SUPPORT COMMAND, LIGHT INFANTRY DIVISION
D0000024.dbf	43-187-30-MTP	6/3/91	ORDNANCE (MAINTENANCE) TROOP, SUPPORT SQUADRON, ARMORED CAVALRY REGIMENT (ACR)
D0000007.dbf	43-209-30-MTP	10/25/90	ORDNANCE (MAINTENANCE) COMPANY, NON-DIVISIONAL, DIRECT SUPPORT (DS), CSG/TAACOM
D0000025.dbf	43-436-30-MTP	11/16/90	HEADQUARTERS DETACHMENT, ORDNANCE (MAINTENANCE) BATTALION, CSG/TAACOM
D0000009.dbf	43-607-30-MTP	11/28/91	ORDNANCE (MAINTENANCE) COMPANY, DIRECT SUPPORT, PATRIOT BRIGADE SUPPORT ELEMENT, AIR DEFENSE SUPPORT COMMAND, THEATER ARMY
D0000010.dbf	43-637-30-MTP	11/16/90	ORDNANCE (MAINTENANCE) COMPANY, LIGHT EQUIPMENT (LEMCO) AND HEAVY EQUIPMENT (HEMCO) GENERAL SUPPORT (GS), TAACOM
D0000026.dbf	43-649-30-MTP	1/22/93	ORDNANCE DETACHMENT, EXPLOSIVE ORDNANCE DISPOSAL
D0000028.dbf	9-527-30-MTP	9/2/93	ORDNANCE DETACHMENT, EXPLOSIVE ORDNANCE DISPOSAL
D0000199.dbf	45-413-30-MTP	9/24/90	MOBILE PUBLIC AFFAIRS DETACHMENT
D0000200.dbf	45-500-10-MTP	9/24/90	PUBLIC AFFAIRS TEAM (PAT)
D0000152.dbf	10-602-MTP	9/22/93	HEADQUARTERS AND HEADQUARTERS COMPANY, PETROLEUM GROUP
D0000117.dbf	10-416-MTP	9/22/93	QUARTERMASTER BATTALION (PETROLEUM PIPELINE AND TERMINAL OPERATING)
D0000118.dbf	10-416-30-MTP	10/27/93	HEADQUARTERS & HQ COMPANY, PETROLEUM GROUP & HQ & HQ CO, QUARTERMASTER BATTALION (PETROLEUM PIPELINE & TERMINAL OPERATING)
D0000027.dbf	10-426-MTP	9/30/93	PETROLEUM SUPPLY BATTALION
D0000119.dbf	10-426-30-MTP	10/27/93	HEADQUARTERS AND HEADQUARTERS DETACHMENT PETROLEUM SUPPLY BATTALION
D0000122.dbf	10-466-MTP	9/16/91	HEADQUARTERS
D0000116.dbf	10-337-30-MTP	12/11/90	QUARTERMASTER AIRDROP EQUIPMENT SUPPORT COMPANY, SUPPLY AND TRANSPORT BATTALION, AIRBORNE DIVISION
D0000124.dbf	10-417-30-MTP	10/27/93	QUARTERMASTER PETROLEUM PIPELINE AND TERMINAL OPERATING COMPANY PETROLEUM SUPPLY COMPANY
D0000120.dbf	10-427-30-MTP	9/30/93	QUARTERMASTER PETROLEUM PIPELINE AND TERMINAL OPERATING COMPANY
D0000121.dbf	10-443-30-MTP	9/22/93	QUARTERMASTER HEAVY/LIGHT AIRDROP SUPPLY COMPANY QM AIRDROP SUPPLY DET QM PARACHUTE PACKING/PARACHUTE REPAIR DET
D0000150.dbf	10-449-30-MTP	10/12/93	AIRDROP EQUIPMENT REPAIR AND SUPPLY COMPANY OR DETACHMENT
D0000028.dbf	10-466-30-MTP	9/16/91	HEADQUARTERS DETACHMENT, QUARTERMASTER BATTALION (WATER SUPPLY)
D0000151.dbf	10-468-30-MTP	9/16/91	QUARTERMASTER CO (WATER SUPPLY) (DS/GS) TACTICAL WATER DIST TEAM (HOSELINE) AND WATER PURIFICATION TEAM (BARGE-MOUNTED ROWPU)

D0000123.dbf	10-469-30-MTP	9/13/91	QUARTERMASTER WATER PURIFICATION DETACHMENT (GS) and QUARTERMASTER WATER PURIFICATION TEAM (12000 GHP)
D0000030.dbf	42-26-MTP	3/29/90	BATTALION HEADQUARTERS, SUPPLY AND TRANSPORT BATTALION, LIGHT INFANTRY, AIRBORNE AND AIR ASSAULT DIVISIONS
D0000035.dbf	42-446-MTP	12/8/92	BATTALION HEADQUARTERS SUPPLY AND SERVICES BATTALION
D0000168.dbf	11-447-30-MTP	11/20/90	BATTALION HEADQUARTERS, SUPPLY AND TRANSPORT BATTALION, LIGHT INFANTRY, AIRBORNE AND AIR ASSAULT DIVISIONS
D0000153.dbf	42-004-30-MTP	2/17/89	SUPPLY COMPANY, FORWARD SUPPORT BATTALION, ARMORED AND MECHANIZED DIVISION
D0000154.dbf	42-007-30-MTP	2/17/89	SUPPLY AND SERVICE COMPANY, MAIN SUPPORT BATTALION, HEAVY DIVISION
D0000029.dbf	42-026-30-MTP	8/8/90	SUPPLY COMPANY, SUPPLY AND TRANSPORT BATTALION, LIGHT INFANTRY, AIRBORNE AND AIR ASSAULT DIVISIONS
D0000155.dbf	42-027-30-MTP	12/11/90	FORWARD SUPPLY COMPANY, SUPPLY AND TRANSPORT BATTALION, LIGHT INFANTRY, AIRBORNE AND AIR ASSAULT DIVISIONS
D0000156.dbf	42-077-30-MTP	5/13/91	SPLY AND TRNSPRT CO, SPRT BN, HVY SEPARATE BDE OR SEPARATE INF BDE OR THEATER DFNSIVE BDE AND SPLY AND TRNSPRT TROOP, SPRT SQDN ARMD CAV REG
D0000031.dbf	42-414-30-MTP	3/29/90	QUARTERMASTER FIELD SERVICE COMPANY (DS), CORPS SUPPORT BATTALION OR SUPPLY AND SERVICES BATTALION (FINAL DRAFT)
D0000032.dbf	42-418-30-MTP	12/8/92	QUARTERMASTER SUPPLY COMPANY (GS), SUPPLY AND SERVICES BATTALION
D0000033.dbf	42-419-30-MTP	12/8/92	QUARTERMASTER REPAIR PARTS SUPPLY COMPANY (GS), SUPPLY AND SERVICES BATTALION
D0000034.dbf	42-427-30-MTP	12/8/92	QUARTERMASTER HEAVY MATERIEL SUPPLY COMPANY (GS), SUPPLY AND SERVICES BATTALION
D0000157.dbf	42-446-30-MTP	12/8/92	HEADQUARTERS DETACHMENT SUPPLY AND SERVICES BATTALION
D0000158.dbf	42-447-30-MTP	12/8/92	QUARTERMASTER SUPPLY COMPANY (DS), CORPS SUPPORT BATTALION OR SUPPLY AND SERVICES BATTALION
DSIG0001.dbf	11-1-MTP	12/1/96	Echelon Corps & Below (Signal), and Battalion and Brigade Commander's Battle Task
D0000098.dbf	11-435-MTP	11/20/90	CORPS AREA SIGNAL BATTALION (MSE)
D0000100.dbf	11-445-MTP	11/20/90	CORPS SUPPORT SIGNAL BATTALION (MSE)
D0000105.dbf	11-XXX-30-MTP	3/16/94	SIGNAL TELECOMMUNICATIONS BATTALION (AREA)
D0000099.dbf	11-437-10-MTP	11/20/90	COMPANIES AND PLATOONS OF THE CORPS AREA SIGNAL BATTALION (MSE)
D0000101.dbf	11-47-10-MTP	9/30/87	COMMAND SIGNAL CENTER PLATOON COMMAND COMMUNICATIONS COMPANY SIGNAL BATTALION LIGHT DIVISION
D0000102.dbf	11-47-11-MTP	9/30/87	COMMAND COMMUNICATIONS SUPPORT PLATOON, COMMAND COMMUNICATIONS COMPANY, SIGNAL BATTALION, LIGHT DIVISION
D0000169.dbf	11-47-13-MTP	9/30/87	DIVISION SUPPORT COMMAND OPERATIONS PLATOON, COMMAND COMMUNICATIONS COMPANY, SIGNAL BATTALION, LIGHT DIVISION
D0000236.dbf	11-47-20-MTP	9/30/87	TACTICAL COMMAND POST SECTION, COMMAND OPERATIONS COMPANY, SIGNAL BATTALION, LIGHT DIVISION
D0000103.dbf	11-48-10-MTP	9/30/87	FORWARD AREA SIGNAL CENTER PLATOON, SIGNAL SUPPORT COMPANY, SIGNAL BATTALION LIGHT DIVISION
D0000104.dbf	11-48-11-MTP	9/30/87	GENERAL PURPOSE PLATOON, SIGNAL SUPPORT COMPANY, SIGNAL BATTALION, LIGHT DIVISION

D0000167.dbf	11-067-30-MTP	11/14/90	COMPANIES AND PLATOONS OF THE DIVISION SIGNAL BATTALION (MSE)
D0000063.dbf	31-807-33-MTP	12/20/90	SPECIAL FORCES COMPANY
D0000064.dbf	33-705-MTP	10/23/89	PSYCHOLOGICAL OPERATIONS BATTALION HEADQUARTERS
D0000065.dbf	33-707-30-MTP	10/23/89	PSYCHOLOGICAL OPERATIONS REGIONAL SUPPORT COMPANY
D0000066.dbf	33-708-30-MTP	10/23/89	PSYCHOLOGICAL OPERATIONS TACTICAL SUPPORT COMPANY
D0000042.dbf	63-422-MTP	9/29/94	HEADQUARTERS CORPS SUPPORT GROUP
D0000079.dbf	63-426-MTP	3/11/92	HEADQUARTERS CORPS SUPPORT BATTALION
D0000043.dbf	63-422-30-MTP	3/11/92	HEADQUARTERS COMPANY CORPS SUPPORT GROUP AND HEADQUARTERS DETACHMENT CORPS SUPPORT BATTALION
D0000080.dbf	63-622-MTP	9/8/93	HEADQUARTERS, AREA SUPPORT GROUP
D0000052.dbf	63-622-30-MTP	9/8/93	HEADQUARTERS COMPANY, AREA SUPPORT GROUP
D0000048.dbf	63-001-MTP	8/25/89	DISCOM HEADQUARTERS/DMMC SUPPORT COMMAND, HEAVY AND MOTORIZED DIVISIONS
D0000072.dbf	63-042-MTP	10/19/89	DISCOM HEADQUARTERS/DMMC SUPPORT COMMAND AIRBORNE AIR ASSAULT AND LIGHT DIVISIONS
D0000050.dbf	63-005-MTP	10/3/88	BATTALION HEADQUARTERS, FORWARD SUPPORT BATTALION, HEAVY AND MOTORIZED DIVISIONS
D0000041.dbf	63-216-MTP	8/18/92	HEADQUARTERS FORWARD SUPPORT BATTALION LIGHT INFANTRY DIVISION
D0000051.dbf	63-065-MTP	12/6/90	HEADQUARTERS SUPPORT SQUADRON ARMORED CAVALRY REGIMENT
D0000074.dbf	63-066-30-MTP	12/6/90	HEADQUARTERS TROOP SUPPORT SQUADRON ARMORED CAVALRY REGIMENT AND HEADQUARTERS COMPANY SUPPORT BATTALION SEPARATE BRIGADES
D0000075.dbf	63-085-MTP	12/6/90	HEADQUARTERS SUPPORT BATTALION HEAVY SEPARATE BRIGADES
D0000077.dbf	63-226-MTP	8/18/92	BATTALION HEADQUARTERS MAIN SUPPORT BATTALION LIGHT INFANTRY DIVISION
D0000076.dbf	63-125-MTP	3/30/89	HEADQUARTERS MAIN SUPPORT BATTALION
D0000049.dbf	63-002-30-MTP	10/3/95	HEADQUARTERS COMPANY, DISCOM SUPPORT COMMAND HEAVY AND MOTORIZED DIVISIONS
D0000071.dbf	63-006-30-MTP	4/7/89	HEADQUARTERS DETACHMENT FORWARD SUPPORT BATTALION, MAIN SUPPORT BATTALION, HEAVY AND MOTORIZED DIVISIONS
D0000073.dbf	63-042-30-MTP	8/25/89	HEADQUARTERS COMPANY, DISCOM AIRBORNE AIR ASSAULT AND LIGHT DIVISIONS
D0000090.dbf	63-146-30-MTP	9/9/93	SUPPLY COMPANY MAIN SUPPORT BATTALION/FORWARD SUPPORT BATTALION AIRBORNE, AIR ASSAULT AND LIGHT INFANTRY DIVISIONS
D0000078.dbf	63-412-30-MTP	10/23/93	COMPANY HQS, SPECIAL TROOPS HQS, THEATER ARMY AREA COMMAND & HQS COMPANY, SPECIAL TROOPS BATTALION, CORPS SUPPORT COMMAND
D0000091.dbf	63-412-75-MTP	9/29/94	SPECIAL TROOPS HEADQUARTERS, THEATER ARMY AREA COMMAND AND HEADQUARTERS, SPECIAL TROOPS BATTALION, CORPS SUPPORT COMMAND
D0000054.dbf	55-1004-70-MTP	11/20/87	DEPLOYMENT SUPPORT BRIGADE
D0000162.dbf	55-62-MTP	9/1/94	HEADQUARTERS, TRANSPORTATION COMPOSITE GROUP
D0000159.dbf	55-601-MTP	1/4/95	HEADQUARTERS, TRANSPORTATION COMMAND
D0000160.dbf	55-603-MTP	1/4/95	THEATER ARMY MOVEMENT CONTROL AGENCY
D0000161.dbf	55-604-MTP	9/1/94	TRANSPORTATION MOVEMENT CONTROL CENTER (CORPS)
D0000058.dbf	55-716-MTP	11/3/93	TRANSPORTATION MOTOR TRANSPORT BATTALION
D0000060.dbf	55-816-MTP	9/10/93	HEADQUARTERS, TRANSPORTATION TERMINAL BATTALION
D0000061.dbf	55-916-MTP	9/28/94	HEADQUARTERS, TRANSPORTATION RAILWAY BATTALION

D0000055.dbf	55-158-30-MTP	10/5/89	TRANSPORTATION MOTOR TRANSPORT COMPANY, SUPPLY AND TRANSPORT BATTALION, AIRBORNE, AIR ASSAULT AND LIGHT DIVISIONS
D0000057.dbf	55-500-30-MTP	9/30/93	TRANSPORTATION WATERCRAFT UNITS
D0000163.dbf	55-716-30-MTP	9/30/93	HQ DETACHMENT TRANS MOTOR TRANS BATT AND TRANS RAILWAY BATT AND HQ CO TRANS TERMINAL BATT
D0000059.dbf	55-717-30-MTP	11/3/93	TRANS COMMAND TRANSPORT CO, TRANS LIGHT TRUCK CO, TRANS LIGHT-MEDIUM TRUCK CO, TRANS HEAVY TRUCK CO, TRANS MOTOR TRANSPORT BN
D0000062.dbf	55-918-30-MTP	5/19/93	TRANSPORTATION RAILWAY ENGINEERING COMPANY
D0000165.dbf	55-919-30-MTP	5/19/93	TRANSPORTATION RAILWAY EQUIPMENT MAINTENANCE COMPANY
D0000166.dbf	55-927-30-MTP	5/19/93	TRANSPORTATION TRAIN OPERATING COMPANY
D0000164.dbf	55-717-10-MTP	11/20/87	TRANSPORTATION TRUCK PLATOON AND SQUAD TRANSPORTATION MOTOR TRANSPORT COMPANY AND DETACHMENT
D0000053.dbf	55-1001-70-MTP	11/20/87	TRANSPORTATION TERMINAL UNIT
D0000998.dbf	6-037-30-MTP (Pending)	4/1/97	FIELD ARTILLERY BATTALION / BATTERY / PLATOON (Pending)

## APPENDIX B - MTP CONVERSION ANNOTATED SOURCE CODE

```

Sub Main()
    ' This code takes a MS Access database of R-TASK MTPs
    ' and strips out the Tasks, Conditions and Standards
    ' for each task within the database and rewrites
    ' that data sequentially into a table of the new database
    ' with a counter added - This counter is the
    ' numerical sequence of the particular task within the database.

    Dim I, J, k As Integer
    Dim db, dbnew As Database
    Dim Tcsfields(9) As Field
    Dim Updatebool As Boolean
    Dim Table, Table1, Tabletc, Tablepm As Recordset
    Dim Parcur, Parnew, Col8cur, Col8new, Linecur, Linenew, Indentnew, Tasknum, Internaltable As String
    Dim Levelcur, Levelnew As Integer
    Dim Artep, BOS, Element, Task, Condition, Standard As String
    Set db = OpenDatabase("f:\fy97\warsim\rci_dat3\checkit.mdb")
    Set db2 = OpenDatabase("f:\fy97\warsim\newtask.mdb")
    Set Table1 = db.OpenRecordset("mtpdir")
    Do Until Table1.EOF
        k = 0
        Internaltable = Left(Table1.Fields(1), 8)
        Set Table = db.OpenRecordset(Internaltable)
        Set Tabletc = db2.CreateTableDef(Internaltable)
        Set Tcsfields(0) = Tabletc.CreateField("BOS", dbText)
        Set Tcsfields(1) = Tabletc.CreateField("Element", dbText)
        Set Tcsfields(2) = Tabletc.CreateField("Tasknumber", dbText)
        Set Tcsfields(3) = Tabletc.CreateField("Task", dbMemo)
        Set Tcsfields(4) = Tabletc.CreateField("Condition", dbMemo)
        Set Tcsfields(5) = Tabletc.CreateField("Standard", dbMemo)
        Set Tcsfields(6) = Tabletc.CreateField("Counter", dbInteger)
        J = 0
        Do While J < 7
            Tabletc.Fields.Append Tcsfields(J)
            J = J + 1
        Loop
        db2.TableDefs.Append Tabletc
        Set Tabletc = db2.OpenRecordset(Internaltable)
        Levelcur = 0
        Levelnew = 0
        Task = ""
        Condition = ""
        Standard = ""
        Element = ""

```

```

BOS = ""
Tasknum = ""
I = 1
Indentnew = ""
Indentcur = ""
Do Until Table.EOF
    Levelnew = Levelcur
    Col8cur = Table.Fields(8)
    Levelcur = Table.Fields(3)
    Parcur = Table.Fields(4)
    Linecur = Table.Fields(5)
    Indentnew = Indentcur
    Indentcur = Table.Fields(7)
    If (Indentnew = "PER" And Indentcur = "CND" Or (Indentnew = "BOS" And Indentcur = "BOS")) Then
        J = J + 1
        Tablects.AddNew
        If BOS <> "" Then Tablects.Fields(0) = BOS
        If Element <> "" Then Tablects.Fields(1) = Element
        If Tasknum <> "" Then Tablects.Fields(2) = Tasknum
        If Task <> "" Then Tablects.Fields(3) = Task
        If Condition <> "" Then Tablects.Fields(4) = Condition
        If Standard <> "" Then Tablects.Fields(5) = Standard
        Tablects.Fields(6) = I
        I = I + 1
    Tablects.Update
End If
If Table.Fields(8) <> "" Then
    If Levelcur = 3 And Parcur = 1 And Linecur = 1 And Table.Fields(7) <> "PER" Then Task = Right(Table.Fields(8),
    Len(Table.Fields(8)) - 5)
    If Levelcur = 3 And Parcur = 1 And Linecur > 1 And Table.Fields(7) <> "PER" Then Task = Task + " " + Table.Fields(8)
    If Levelcur <= 3 Then Tasknum = Table.Fields(2)
    If Levelcur <= 3 Then Tasknum = Table.Fields(2)
    If Left(Table.Fields(8), 4) = "BOS:" Then BOS = Right(Table.Fields(8), Len(Table.Fields(8)) - 4)
    If Left(Table.Fields(8), 8) = "ELEMENT:" Then Element = Right(Table.Fields(8), Len(Table.Fields(8)) - 8)
    If Levelcur = 3 And Parcur = 2 And Linecur = 1 And Left(Table.Fields(8), 10) = "CONDITION:" Then Condition =
    Right(Table.Fields(8), Len(Table.Fields(8)) - 10)
    If Levelcur = 3 And Parcur = 2 And Linecur = 1 And Left(Table.Fields(8), 11) = "CONDITIONS:" Then Condition =
    Right(Table.Fields(8), Len(Table.Fields(8)) - 11)
    If Levelcur = 3 And Parcur = 2 And Linecur > 1 Then Condition = Condition + " " + Table.Fields(8)
    If Levelcur = 3 And Parcur = 3 And Linecur = 1 And Left(Table.Fields(8), 9) = "STANDARD:" Then Standard =
    Right(Table.Fields(8), Len(Table.Fields(8)) - 9)
    If Levelcur = 3 And Parcur = 3 And Linecur = 1 And Left(Table.Fields(8), 14) = "TASK STANDARD:" Then Standard =
    Right(Table.Fields(8), Len(Table.Fields(8)) - 14)
    If Levelcur = 3 And Parcur = 3 And Linecur = 1 And Left(Table.Fields(8), 15) = "TASK STANDARDS:" Then Standard =
    Right(Table.Fields(8), Len(Table.Fields(8)) - 15)
    If Levelcur = 3 And Parcur = 3 And Linecur > 1 Then Standard = Standard + " " + Table.Fields(8)
End If
Table.MoveNext
Loop

```

```

' Write out the last record
,
Tabletcs.AddNew
If BOS <> "" Then Tabletcs.Fields(0) = BOS
If Element <> "" Then Tabletcs.Fields(1) = Element
If Tasknum <> "" Then Tabletcs.Fields(2) = Tasknum
If Task <> "" Then Tabletcs.Fields(3) = Task
If Condition <> "" Then Tabletcs.Fields(4) = Condition
If Standard <> "" Then Tabletcs.Fields(5) = Standard
BOS = ""
Task = ""
Condition = ""
Standard = ""
Element = ""
Tabletcs.Fields(6) = I
Tabletcs.Update
Table.Close
Table1.MoveNext
Loop
Table1.Close
Tabletcs.Close
db.Close
db2.Close
End Sub

Sub Main()
' This code takes a MS Access database of R-TASK MTPs
' and strips out the Steps and Performance Measures for every task
' within the database and rewrites that data sequentially into a table of the new
' database with the same counter as the previous program but with
' each step and performance measure also sequenced.

Dim I, J, K, L, M, N, O, Z, A As Integer
Dim db, dbnew As Database
Dim Tcsfields(9) As Field
Dim Updatebool As Boolean
Dim Table, Table1, Tabletcs, Tablepm As Recordset
Dim Parcur, Parprev, Col8cur, Col8new, Artepdate, Tasknum, Internaltable, Indentnew, Indentcur, Tasktype As String
Dim Levelcur, Levelnew, Lineprev, Linecur, Linenew As Integer
Dim Artep, BOS, Element, Task, Condition, Standard, Texttype As String
Set db = OpenDatabase("f:\fy97\warsim\rci_dat3\checkit.mdb")
Set db2 = OpenDatabase("f:\fy97\warsim\newpm.mdb")
Set Table1 = db.OpenRecordset("mtpdir")
Do Until Table1.EOF
    Internaltable = Left(Table1.Fields(1), 8)

```

```

Set Table = db.OpenRecordset(InternalTable)
Set Tablecs = db2.CreateTableDef(InternalTable)
Set Tcsfields(0) = Tablecs.CreateField("Tasknumber", dbText)
Set Tcsfields(1) = Tablecs.CreateField("Step", dbText)
Set Tcsfields(2) = Tablecs.CreateField("Type", dbInteger)
Set Tcsfields(3) = Tablecs.CreateField("Narrative", dbMemo)
Set Tcsfields(4) = Tablecs.CreateField("WMPTaskReference", dbInteger)
Set Tcsfields(5) = Tablecs.CreateField("WMPStepReference", dbInteger)
Set Tcsfields(6) = Tablecs.CreateField("WMPPMReference", dbInteger)
Set Tcsfields(7) = Tablecs.CreateField("WMPSubPMReference", dbInteger)
J = 0
Do While J < 8
    Tablecs.Fields.Append Tcsfields(J)
    J = J + 1
Loop
db2.TableDefs.Append Tablecs
Set Tablecs = db2.OpenRecordset(InternalTable)
Levelcur = 0
Indentnew = ""
Indentcur = ""
Levelnew = 0
L = 0
Z = -1
M = 0
N = 0
I = 1
Do Until Table.EOF
    Parprev = Parcur
    Indentnew = Indentcur
    Indentcur = Table.Fields(7)
    Levelnew = Levelcur
    Col8cur = Table.Fields(8)
    Levelcur = Table.Fields(3)
    Parcur = Table.Fields(4)
    Lineprev = Linecur
    Linecur = Table.Fields(5)
    If Table.Fields(7) = "BOS" And L > 0 Then Z = L
    If Table.Fields(7) = "TSK" And Table.Fields(5) = "1" Then
        L = L + 1
        M = 0
    End If
    If (Levelnew >= 4 And Linecur = 1 And Table.Fields(7) = "PER") Or (Indentnew = "PER" And Indentcur = "BOS") Then ' write our the
record
        Tablecs.AddNew
        Z = Z + 1
        If Tasknum <> "" Then
            K = 6
            Do While K < 30
                If (Mid(Tasknum, K, 9)) = "
                    Tablecs.Fields(0) = Left(Tasknum, K - 1)

```

```

Tabletcs.Fields(1) = Right(Tasknum, Len(Tasknum) - (K + 8))
K = 51
ElseIf (Mid(Tasknum, K, 8)) = " " Then
    Tabletcs.Fields(0) = Left(Tasknum, K - 1)
    Tabletcs.Fields(1) = Right(Tasknum, Len(Tasknum) - (K + 7))
K = 51
ElseIf (Mid(Tasknum, K, 7)) = " " Then
    Tabletcs.Fields(0) = Left(Tasknum, K - 1)
    Tabletcs.Fields(1) = Right(Tasknum, Len(Tasknum) - (K + 6))
K = 51
ElseIf (Mid(Tasknum, K, 6)) = " " Then
    Tabletcs.Fields(0) = Left(Tasknum, K - 1)
    Tabletcs.Fields(1) = Right(Tasknum, Len(Tasknum) - (K + 5))
K = 51
ElseIf (Mid(Tasknum, K, 5)) = " " Then
    Tabletcs.Fields(0) = Left(Tasknum, K - 1)
    Tabletcs.Fields(1) = Right(Tasknum, Len(Tasknum) - (K + 4))
K = 51
ElseIf (Mid(Tasknum, K, 4)) = " " Then
    Tabletcs.Fields(0) = Left(Tasknum, K - 1)
    Tabletcs.Fields(1) = Right(Tasknum, Len(Tasknum) - (K + 3))
K = 51
ElseIf (Mid(Tasknum, K, 3)) = " " Then
    Tabletcs.Fields(0) = Left(Tasknum, K - 1)
    Tabletcs.Fields(1) = Right(Tasknum, Len(Tasknum) - (K + 2))
K = 51
Else
    K = K + 1
End If
End If
Loop
End If
If Tasktype <> " " Then Tabletcs.Fields(2) = Tasktype
If Texttype <> " " Then Tabletcs.Fields(3) = Texttype
Tabletcs.Fields(4) = L
Tabletcs.Fields(5) = M
Tabletcs.Fields(6) = N
Tabletcs.Fields(7) = O
I = I + 1
Tabletcs.Update
End If
If Table.Fields(8) <> " " And Table.Fields(7) = "PER" Then
    Tasknum = Table.Fields(2)
    If Levelcur >= 4 Then Tasktype = Levelcur
    If Levelcur >= 4 And Linecur = 1 Then Texttype = Table.Fields(8)
    If Levelcur >= 4 And Linecur > 1 Then: Texttype = Texttype + " " + Table.Fields(8)
End If
If (Table.Fields(3) = "4" And Table.Fields(5) = "1" And Table.Fields(7) = "PER") Then
    M = M + 1
    N = 0
    O = 0

```

```

End If
If (Table.Fields(3) = "5" And Table.Fields(5) = "1" And Table.Fields(7) = "PER") Then
    N = N + 1
    O = 0
End If
If (Table.Fields(3) = "6" And Table.Fields(5) = "1" And Table.Fields(7) = "PER") Then
    O = O + 1
End If
Table.MoveNext
Loop
Table.Close
Table1.MoveNext
,
, Adding last subtask here
,
Tabletcs.AddNew
If Tasknum <> " " Then
    K = 6
    Do While K < 30
        If (Mid(Tasknum, K, 9)) = " " Then
            Tabletcs.Fields(0) = Left(Tasknum, K - 1)
            Tabletcs.Fields(1) = Right(Tasknum, Len(Tasknum) - (K + 8))
            K = 51
        ElseIf (Mid(Tasknum, K, 8)) = " " Then
            Tabletcs.Fields(0) = Left(Tasknum, K - 1)
            Tabletcs.Fields(1) = Right(Tasknum, Len(Tasknum) - (K + 7))
            K = 51
        ElseIf (Mid(Tasknum, K, 7)) = " " Then
            Tabletcs.Fields(0) = Left(Tasknum, K - 1)
            Tabletcs.Fields(1) = Right(Tasknum, Len(Tasknum) - (K + 6))
            K = 51
        ElseIf (Mid(Tasknum, K, 6)) = " " Then
            Tabletcs.Fields(0) = Left(Tasknum, K - 1)
            Tabletcs.Fields(1) = Right(Tasknum, Len(Tasknum) - (K + 5))
            K = 51
        ElseIf (Mid(Tasknum, K, 5)) = " " Then
            Tabletcs.Fields(0) = Left(Tasknum, K - 1)
            Tabletcs.Fields(1) = Right(Tasknum, Len(Tasknum) - (K + 4))
            K = 51
        ElseIf (Mid(Tasknum, K, 4)) = " " Then
            Tabletcs.Fields(0) = Left(Tasknum, K - 1)
            Tabletcs.Fields(1) = Right(Tasknum, Len(Tasknum) - (K + 3))
            K = 51
        ElseIf (Mid(Tasknum, K, 3)) = " " Then
            Tabletcs.Fields(0) = Left(Tasknum, K - 1)
            Tabletcs.Fields(1) = Right(Tasknum, Len(Tasknum) - (K + 2))
            K = 51
        Else
            K = K + 1
        End If
    End If

```

```

Loop
End If
If Tasktype <> "" Then Tabletcsc.Fields(2) = Tasktype
If Texttype <> "" Then Tabletcsc.Fields(3) = Texttype
Tabletcsc.Fields(4) = L
Tabletcsc.Fields(5) = M
Tabletcsc.Fields(6) = N
Tabletcsc.Fields(7) = 0
I = I + 1
Tabletcsc.Update
,
, added last record
,
Loop
Table1.Close
Tabletcsc.Close
db.Close
db2.Close
End Sub

```

## APPENDIX C - SOURCE CODE FOR CREATION OF USER ACCOUNTS (AUTH.EXE)

```

Sub CGI_Main()
Dim xx As Integer
Dim Pass1, Pass2, Logname, name, Unit, Phone, Email, Login, Event, Address As String
Dim Db As Database
Dim Logbool, Errorflag, Expbool As Boolean
Dim Table, Table1 As Recordset
Dim Tfield As Field
Dim Mytabledef As TableDef
Dim i As Integer
Errorflag = False
sSelector = UCASE$(Mid$(CGI_LogicalPath, 2)) ' Remove leading "/"
Send ("Content-type: text/html")
Send ("")
Send ("<HTML><HEAD><TITLE><TITLE> User Request Results</CENTER></TITLE>")
Send ("<H1> User Request Results </H1></HEAD>")
Send ("<BODY TEXT="#000000" BGCOLOR="#00FFFF" LINK="#0000EE" VLINK="#551A8B" ALINK="#FF0000">")

Send ("<P>")
,
, Password and Expert Checking
,
i = 0
Do While i < CGI_NumFormTuples
If CGI_FormTuples(i).key = "Pass1" Then
Pass1 = CGI_FormTuples(i).value
End If
If CGI_FormTuples(i).key = "Pass2" Then
Pass2 = CGI_FormTuples(i).value
End If
If CGI_FormTuples(i).key = "Expert" Then
Event = CGI_FormTuples(i).value
End If
If CGI_FormTuples(i).key = "Name" Then
name = CGI_FormTuples(i).value
End If
If CGI_FormTuples(i).key = "Unit" Then
Unit = CGI_FormTuples(i).value
End If
If CGI_FormTuples(i).key = "Phone" Then
Phone = CGI_FormTuples(i).value
End If
If CGI_FormTuples(i).key = "Email" Then
Email = CGI_FormTuples(i).value
End If
If CGI_FormTuples(i).key = "Loginname" Then
Login = CGI_FormTuples(i).value

```

```

End If
If CGI_FormTuples(i).key = "Address" Then
    Address = CGI_FormTuples(i).value
End If
i = i + 1
Loop
Send ("<h3>")

If Event = "" Then Send ("You must select at least one field as a Subject Matter Expert - Please try again!<P>")
If Name = "" Then Send ("I must have your name, Please try again!<P>")
If Unit = "" Then Send ("I must have your unit, Please try again!<P>")
If Phone = "" Then Send ("I must have your phone number, Please try again!<P>")
If Email = "" Then Send ("I must have your email address, Please try again!<P>")
If Login = "" Then Send ("You must provide a login name, Please try again!<P>")
If Address = "" Then Send ("You must provide an address, Please try again!<P>")
If Pass1 = "" Or Pass2 = "" Then Send ("Passwords must be entered. Please try Again!<P>")
If Pass1 <> Pass2 Then Send ("Passwords do not match - Please try again!<P>")

' Userid checking
i = 0
Do While i < CGI_NumFormTuples
    If CGI_FormTuples(i).key = "Loginname" Then
        Logname = CGI_FormTuples(i).value
    End If
    i = i + 1
Loop
Set Db = OpenDatabase("c:\netscape\server\docs\database\users.mdb")
Set Table = Db.OpenRecordset("data")
Logbool = True
Do While Not Table.EOF
    If Table.Fields(6) = Logname And Logbool = True Then
        Send ("Sorry, someone has already requested that login name - choose another <P>")
        Logbool = False
    End If
    Table.MoveNext
Loop
If Event <> "" And name <> "" And Unit <> "" And Phone <> "" And Email <> "" And Address <> "" And Login <> "" And Pass1 <> "" And
Pass2 <> "" And Pass1 = Pass2 And Logbool = True Then
    Table.AddNew
    i = 0
    Do While i < CGI_NumFormTuples
        If CGI_FormTuples(i).key = "Name" Then
            Table.Fields(0) = CGI_FormTuples(i).value
        End If
        If CGI_FormTuples(i).key = "Rank" Then
            Table.Fields(1) = CGI_FormTuples(i).value
        End If
    
```

```

If CGI_FormTuples(i).key = "Unit" Then
    Table.Fields(2) = CGI_FormTuples(i).value
End If
If CGI_FormTuples(i).key = "Address" Then
    Table.Fields(3) = CGI_FormTuples(i).value
End If
If CGI_FormTuples(i).key = "Phone" Then
    Table.Fields(4) = CGI_FormTuples(i).value
End If
If CGI_FormTuples(i).key = "Email" Then
    Table.Fields(5) = CGI_FormTuples(i).value
End If
If CGI_FormTuples(i).key = "Loginname" Then
    Table.Fields(6) = CGI_FormTuples(i).value
End If
If CGI_FormTuples(i).key = "Pass1" Then
    Table.Fields(7) = CGI_FormTuples(i).value
End If
If CGI_FormTuples(i).key = "Pass2" Then
    Table.Fields(8) = CGI_FormTuples(i).value
End If
If CGI_FormTuples(i).key = "Expert" Then
    Table.Fields(9) = CGI_FormTuples(i).value
End If
If CGI_FormTuples(i).key = "Expert_1" Then
    Table.Fields(10) = CGI_FormTuples(i).value
End If
If CGI_FormTuples(i).key = "Expert_2" Then
    Table.Fields(11) = CGI_FormTuples(i).value
End If
If CGI_FormTuples(i).key = "Expert_3" Then
    Table.Fields(12) = CGI_FormTuples(i).value
End If
If CGI_FormTuples(i).key = "Expert_4" Then
    Table.Fields(13) = CGI_FormTuples(i).value
End If
If CGI_FormTuples(i).key = "Expert_5" Then
    Table.Fields(14) = CGI_FormTuples(i).value
End If
If CGI_FormTuples(i).key = "Expert_6" Then
    Table.Fields(15) = CGI_FormTuples(i).value
End If
If CGI_FormTuples(i).key = "Expert_7" Then
    Table.Fields(16) = CGI_FormTuples(i).value
End If
If CGI_FormTuples(i).key = "Expert_8" Then
    Table.Fields(17) = CGI_FormTuples(i).value
End If
If CGI_FormTuples(i).key = "Comments" Then
    Table.Fields(18) = CGI_FormTuples(i).value

```

```

End If
i = i + 1
Loop
Table.Fields(19) = Now
Table.Fields(20) = CGI_RemoteAddr
Table.Update

Table.Close
Send ("Congratulations, you have successfully entered the required information. It may take 1-2 days to get your account
established.")
Send ("If you need to start immediately please contact <img src=\"/graphics/mailbox.gif\"><A
HREF=\"mailto:fr0161@exmail.usma.edu\">MAJ Bob Phelan</a>")
Send (" <p> You can also call him at (DSN) 688-5941; (914) 938-5941.")
Send ("<p>")
Send ("<p>")
Send ("<p>")
Send ("<A HREF=\"http://www.orcen.usma.edu/\">Operations Research Center</a>")

Expbool = False
Set Table1 = Db.OpenRecordset("Expertise")
i = 0
Do While i < CGI_NumFormTuples
If Left(CGI_FormTuples(i).key, 3) = "exp" Then
If Expbool = False Then Expbool = True

Table1.AddNew
Table1.Fields(0) = Login
Table1.Fields(1) = CGI_FormTuples(i).key
Table1.Update

End If
i = i + 1
Loop
Table1.Close

Else
Send (" <blink> Please go 'BACK' and reenter the required information.</blink> <p>")
Send ("If you continue to have problems, please contact <img src=\"/graphics/mailbox.gif\"> <A
HREF=\"mailto:fr0161@exmail.usma.edu\">MAJ Bob Phelan</a>")
Send (" <p> You can also call him at (DSN) 688-5941; (914) 938-5941.")
Send ("<p>")
Send ("<p>")
Send ("<p>")
Send ("<A HREF=\"http://www.orcen.usma.edu/\">Operations Research Center</a>")

End If

Db.Close
Send ("</h1>")

```

End Sub

## APPENDIX D - SOURCE CODE FOR DISPLAY OF TRAINING AUDIENCE TASKS (MAPPING.EXE)

```

Sub CGI_Main()
    Dim Gotit As Boolean
    Dim xx, i, X, Y, l, m, px, Session, Pos, Totalrecs As Integer
    Dim mptext, Listtype, MTPNAME, Temptask, Lastelement, Preference, Mtptable As String
    Dim Db, Dbl, Userdb, PTADB As Database
    Dim Table, Table1, Usertable, PTAXs As Recordset
    sSelector = UCASE$(Mid$(CGI_LogicalPath, 2)) ' Remove leading "/"
    Send ("Content-type: text/html")
    Send ("")
    px = FreeFile

    l = Len(CGI_QueryString)
    i = 1
    Do While i < l + 1
        If Mid(CGI_QueryString, i, 1) = "," Then
            m = i
            i = l + 1
        Else
            i = i + 1
        End If
    Loop
    Set Db = OpenDatabase("c:\netscape\server\docs\input\database\mtptask.mdb")
    Set PTADB = OpenDatabase("c:\netscape\server\docs\input\database\mtptpsc.mdb")

    ' Working here on PTA Xs

    Set PTAXs = PTADB.OpenRecordset("PTA_Xs", dbReadOnly)
    Set Userdb = OpenDatabase("c:\netscape\server\docs\database\users.mdb")
    Set Usertable = Userdb.OpenRecordset("Data")
    Usertable.LockEdits = False

    ' Get the user's session number
    i = 0
    Preference = ""
    Mtptable = "x"

    Do While i < CGI_NumFormTuples
        If CGI_FormTuples(i).key = "mtpselect" Then Mtptable = Left(CGI_FormTuples(i).value, 8)
        If CGI_FormTuples(i).key = "preference" Then Preference = CGI_FormTuples(i).value
        i = i + 1
    Loop
    mptext = Mtptable

```

```

UserTable.MoveNextFirst
Do Until UserTable.EOF
If UserTable.Fields(6) = CGI.AuthUser Then
    Session = UserTable.Fields(22) + 1
    UserTable.Edit
    UserTable.Fields(22) = Session
    UserTable.Fields(23) = Now
    UserTable.Update
    UserTable.MoveLast
End If
UserTable.MoveNextNext
Loop
UserTable.Close
Userdb.Close
Preference = "task"

X = Val(Left(CGI_QueryString, m - 1))
Y = Val(Right(CGI_QueryString, 1 - m))
' set the table value as mtptext
If mtptext = "x" Then
    ' 14 Apr 97 - Changed to the new div/corps mtp (pending)
    If X > 338 And Y > 49 And X < 384 And Y < 78 Then mtptext = "D0000001"
    If X > 338 And Y > 49 And X < 384 And Y < 78 Then mtptext = "D0000099"
    If X > 337 And Y > 120 And X < 384 And Y < 152 Then mtptext = "D0000210"
    If X > 397 And Y > 120 And X < 443 And Y < 153 Then mtptext = "D0000159"
    If X > 478 And Y > 120 And X < 522 And Y < 151 Then mtptext = "D0000080"
    If X > 130 And Y > 206 And X < 176 And Y < 238 Then mtptext = "D0000240"
    If X > 191 And Y > 206 And X < 237 And Y < 238 Then mtptext = "D0000081"
    If X > 337 And Y > 206 And X < 382 And Y < 238 Then mtptext = "D0000250"
    If X > 579 And Y > 206 And X < 627 And Y < 238 Then mtptext = "D0000247"
    If X > 69 And Y > 295 And X < 116 And Y < 329 Then mtptext = "D0000192"
    If X > 131 And Y > 295 And X < 177 And Y < 329 Then mtptext = "D0000239"
    If X > 192 And Y > 295 And X < 239 And Y < 329 Then mtptext = "D0000001"
    If X > 251 And Y > 295 And X < 300 And Y < 329 Then mtptext = "D0000252"
    If X > 336 And Y > 295 And X < 385 And Y < 329 Then mtptext = "D0000238"
    If X > 397 And Y > 295 And X < 445 And Y < 329 Then mtptext = "D0000037"
    If X > 480 And Y > 295 And X < 527 And Y < 329 Then mtptext = "D0000105"
    If X > 579 And Y > 295 And X < 628 And Y < 329 Then mtptext = "D0000248"
    If X > 69 And Y > 383 And X < 117 And Y < 417 Then mtptext = "D0000189"
    If X > 192 And Y > 383 And X < 239 And Y < 417 Then mtptext = "D0000005"
    If X > 251 And Y > 383 And X < 299 And Y < 417 Then mtptext = "D0000257"
    If X > 397 And Y > 383 And X < 445 And Y < 417 Then mtptext = "D0000039"
    If X > 478 And Y > 383 And X < 525 And Y < 417 Then mtptext = "D0000098"
    If X > 579 And Y > 383 And X < 627 And Y < 417 Then mtptext = "D0000246"
    If X > 68 And Y > 472 And X < 118 And Y < 505 Then mtptext = "D0000197"
    If X > 191 And Y > 472 And X < 240 And Y < 505 Then mtptext = "D0000003"
    If X > 397 And Y > 472 And X < 445 And Y < 505 Then mtptext = "D0000011"
    If X > 477 And Y > 472 And X < 526 And Y < 505 Then mtptext = "D0000100"
    If X > 580 And Y > 472 And X < 628 And Y < 505 Then mtptext = "D0000051"

```

```

If X > 192 And Y > 560 And X < 239 And Y < 593 Then mtptext = "D0000218"
If X > 69 And Y > 686 And X < 118 And Y < 719 Then mtptext = "D0000125"
If X > 130 And Y > 686 And X < 178 And Y < 719 Then mtptext = "D0000108"
If X > 397 And Y > 686 And X < 444 And Y < 719 Then mtptext = "D0000042"
If X > 476 And Y > 686 And X < 524 And Y < 719 Then mtptext = "D0000048"
If X > 579 And Y > 686 And X < 628 And Y < 719 Then mtptext = "D0000072"
If X > 70 And Y > 774 And X < 118 And Y < 807 Then mtptext = "D0000125"
If X > 129 And Y > 774 And X < 178 And Y < 807 Then mtptext = "D0000109"
If X > 192 And Y > 774 And X < 239 And Y < 807 Then mtptext = "D0000068"
If X > 251 And Y > 774 And X < 299 And Y < 807 Then mtptext = "D0000018"
If X > 337 And Y > 774 And X < 385 And Y < 807 Then mtptext = "D0000075"
If X > 397 And Y > 774 And X < 445 And Y < 807 Then mtptext = "D0000079"
If X > 478 And Y > 774 And X < 525 And Y < 807 Then mtptext = "D0000050"
If X > 447 And Y > 863 And X < 525 And Y < 896 Then mtptext = "D0000076"
If X > 579 And Y > 863 And X < 627 And Y < 896 Then mtptext = "D0000077"
End If
Set Table = Db.OpenRecordset("MTPDIR", dbOpenSnapshot)
Table.MoveFirst
Lastelement = ""

If mtptext = "x" Then
    Send ("<HTML><CENTER><TITLE>Training Audience Selection </TITLE></CENTER>")
    Send ("<BODY TEXT=#000000" BGCOLOR="#00FFFF" LINK="#0000EE" VLINK="#551A8B" ALINK="#FF0000">")

    Send ("<Center><H1>Select the Training Audience (MTP) you wish to work with:</H1></center><p>")
    Send ("<FORM METHOD="POST" ACTION="/win.cgi/mapping.exe?0,0">")
    Send ("<INPUT TYPE="HIDDEN" NAME="MTPTABLE" VALUE="" & mtptext & ""> ")
    Send ("<INPUT TYPE="HIDDEN" NAME="SESSION" VALUE="" & Session & ""> ")
    Send ("<TABLE BORDER=1>")
    Do Until Table.EOF
        Send ("<TR><TD>" & Table.Fields(2) & "<br>" & Table.Fields(4) & "</TD><TD>" & Table.Fields(5) & "</td><td><INPUT type = "radio" name = "mtpselect" value=" " & " & Table.Fields(1) & " & "></TD></TR>")
        Table.MoveNext
    Loop
    Send ("</table>")
    Table.Close
    Send ("<p>")
    Send ("<P><Center><INPUT TYPE=SUBMIT VALUE= "Select Tasks to Train" ></Center>")
    Send ("</form>")
Else
    Do Until Table.EOF
        If Left(Table.Fields(1), 8) = mtptext Then MTPNAME = Table.Fields(8)
        Table.MoveNext
    Loop

```

```
' Set Table = Db.OpenRecordset(mtpText, dbOpenSnapshot)
' Set Table1 = Db1.OpenRecordset(mtpText, dbOpenSnapshot)
Send ("<HTML><CENTER><TITLE> Training Audience Tasks for " & MTPNAME & " </TITLE></CENTER>")
Send ("<BODY TEXT=""#000000"" bgcolor=""#00FFFF"" LINK=""#0000EE"" VLINK=""#551A8B"" ALINK=""#FF0000"">")
Send ("<br><H2><Center>" + MTPNAME + "</center></h2><p>")
Send ("<center><b>Select all of the Training tasks that will be trained within WARSIM.</B></center>")
Send ("<center><When complete, press the Submit button at the bottom of this screen.</center><p>")
Send ("<FORM METHOD=""POST"" ACTION= "/wincgi/makeru.exe">")
' Send ("<FORM METHOD=""POST"" ACTION= "/wincgi/pataasks.exe">")
Send ("<INPUT TYPE=""HIDDEN"" NAME=""MPTABLE"" VALUE="" & mtpText & "">")
Send ("<INPUT TYPE=""HIDDEN"" NAME=""MTPNAME"" VALUE="" & MTPNAME & ""> ")
Send ("<INPUT TYPE=""HIDDEN"" NAME=""SESSION"" VALUE="" & Session & ""> ")
Send ("<INPUT TYPE=""HIDDEN"" NAME=""first"" VALUE=""time""> ")
' Set PTAXS = PTADB.OpenRecordset("SELECT DISTINCTROW PTA_Xs.UserId, PTA_Xs.Session, PTA_Xs.MTP, PTA_Xs.Tasknumber, PTA_Xs.Update,
PTA_Xs.Rating, PTA_Xs.Comments From PTA_Xs WHERE ((PTA_Xs.UserId Like " & Chr(39) & CGI_AuthUser & Chr(39) & ") AND (PTA_Xs.MTP Like "
& Chr(39) & Mptext & Chr(39) & "))ORDER BY PTA_Xs.Tasknumber;")
' Send ("SELECT DISTINCTROW PTA_Xs.UserId, PTA_Xs.Session, PTA_Xs.MTP, PTA_Xs.Tasknumber, PTA_Xs.Update, PTA_Xs.Rating, PTA_Xs.Comments
From PTA_Xs WHERE ((PTA_Xs.UserId Like " & Chr(39) & CGI_AuthUser & Chr(39) & ") AND (PTA_Xs.MTP Like " & Chr(39) & Mptext & Chr(39) &
")) ORDER BY PTA_Xs.Tasknumber;")
' Send ("<p>")
' xx = FreeFile
' Open "c:\netscape\server\docs\input\temp\tst13.txt" For Output As #xx
' Print #xx, MTPNAME
' Close #xx
'
Set PTAXS = PTADB.OpenRecordset("SELECT DISTINCTROW PTA_Xs.UserId, PTA_Xs.Session, PTA_Xs.MTP, PTA_Xs.Tasknumber, PTA_Xs.Update,
PTA_Xs.Rating, PTA_Xs.Comments From PTA_Xs WHERE ((PTA_Xs.UserId Like " & Chr(39) & CGI_AuthUser & Chr(39) & ") AND (PTA_Xs.MTP Like "
& Chr(39) & mtpText & Chr(39) & "))ORDER BY PTA_Xs.Tasknumber;")
' Send ("<p>")
' PTAXS.MoveFirst
TotalRecs = 0
If PTAXS.BOF = True And PTAXS.EOF = True Then TotalRecs = -1
'
Send ("<TABLE BORDER=1>")
Send ("<TR><TD><B><H3><Center>Task Number</center></H3></B></TD><B><H3><Center>Task <br> Conditions<br>
standards</center></H3></B></TD><B><H3><Center>PTA TASK?</center></H3></B></TD></TR>")
```

63

```

If Table.Fields(1) <> "" And Table.Fields(1) <> Lastelement Then Send ("<tr><td></td><td><font size=+1><B>ELEMENT: </B>" +
Table.Fields(1) + "</font></td></tr>")
Lastelement = Table.Fields(1)

If Table.Fields(2) <> "" Then
Send ("<TR><TD><font size=+1>" + Table.Fields(2) + "</font></TD><TD>")
Else
Send ("<TR><TD></TD><TD>")
End If

If Table.Fields(3) <> "" Then

TempTask = Table.Fields(3)
Pos = Instr(1, TempTask, "(", 1)
,
, Add the following line to all abbreviated tasks procedures
,

If Pos = 0 Then Pos = Len(Table.Fields(3)) + 1
TempTask = Left$(TempTask, Pos - 1)
Send ("<B><font size=+1>TASK: </font></B>" + TempTask)
,
, This is where I add the drill button to provide more detail about a task
, Need to think about this one
,

End If
,
If Table.Fields(4) <> "" And Preference <> "task" Then Send ("<br><B><font size=+1>CONDITIONS: </font></B>" + Table.Fields(4))
If Table.Fields(5) <> "" And Preference <> "task" Then Send ("<br><B><font size=+1>STANDARDS: </font></B>" + Table.Fields(5))
If Preference = "all" Then
Table1.MoveFirst
Do Until Table1.EOF
If Table1.Fields(4) = Table.Fields(6) And Table1.Fields(2) = 4 Then Send ("<br>" & Table1.Fields(3))
Table1.MoveNext
Loop
End If
,
, Here is where I figure out is a guy X'd the block
,

If Table.Fields(6) <> "" Then Send ("<br></font></td><td><Center><INPUT type = "checkbox" name = "mtpselect" value=" " +
Str(Table.Fields(6)) + " " ")
Gotit = False

If Totalrecs <> -1 Then

```

```

PTAxs.MoveFirst
Do Until PTAxs.EOF
    If PTAxs.Fields(0) = CGI_AuthUser And Left(PTAxs.Fields(2), 8) = mtpText And Val(PTAxs.Fields(3)) = Val(Table.Fields(6)) Then
        Send ("checked"</center></TD></TR>")
        Gotit = True
        PTAxs.MoveLast
    End If
    PTAxs.MoveNext
Loop
End If
If Gotit = False Then
    Send ("></center></TD></TR>")
End If

Table.MoveNext
Loop
Send ("</table>")
Send ("<P><Center><INPUT TYPE=SUBMIT VALUE= "Evaluate Tasks" ><INPUT TYPE= "RESET" VALUE= "Clear Current
Changes"></Center>")
Send ("</form>")
Table.Close
PTAxs.Close
PTAdb.Close
End If

Send ("</HTML>")
Send ("</BODY>")
Db.Close
End Sub

```

## APPENDIX E - SOURCE CODE FOR DISPLAY OF TRAINING AUDIENCE TASKS (UPDTPTA.EXE)

```

Sub CGI_Main()
    Dim Filenumber, Pos, i As Integer
    Dim RUBool, PTAXSBOOL, rudatabool, Tpscbol As Boolean
    Dim Db, Db1, TPSCdb, RUdb As Database
    Dim Table, Table1, TPSCData, RUdata As Recordset
    Dim Rating, tempfile, Teststring, Comments, Prodpipe, Prodpipe2, temptask As String
    Dim Nextrec, xx As Integer
    Dim Currentfield2, Currentfield4, Currentfield5, Currentfield6 As Integer
    i = 0
    Do While i < CGI_NumFormTuples
        If CGI_FormTuples(i).key = "MTPNAME" Then Mtpname = CGI_FormTuples(i).value
        If CGI_FormTuples(i).key = "MPTTABLE" Then MPTTable = CGI_FormTuples(i).value
        If CGI_FormTuples(i).key = "SESSION" Then Session = CGI_FormTuples(i).value
        If CGI_FormTuples(i).key = "TASKNUM" Then Tasknum = CGI_FormTuples(i).value
        i = i + 1
    Loop
    Set Db = OpenDatabase("c:\netscape\server\docs\input\database\mtptask.mdb")
    Set Db1 = OpenDatabase("c:\netscape\server\docs\input\database\mtppm.mdb")
    Set Table = Db.OpenRecordset(MPTTable, dbOpenSnapshot)
    Set Table1 = Db1.OpenRecordset(MPTTable, dbOpenSnapshot)
    Set TPSCdb = OpenDatabase("c:\netscape\server\docs\input\database\mtptpsc.mdb")
    Set RUdb = OpenDatabase("c:\netscape\server\docs\input\database\mtpru.mdb")

    Set TPSCData = TPSCdb.OpenRecordset("SELECT DISTINCTROW " & MPTTable & ".USERID, " & MPTTable & ".Session, " & MPTTable &
    ".Tasknumber, " & MPTTable & ".Stepnumber, " & MPTTable & ".PMNumber, " & MPTTable & ".Rating, " & MPTTable & ".ProdPipe, " & MPTTable
    & ".Comments From " & MPTTable & " WHERE ((" & MPTTable & ".USERID Like " & Chr(39) & CGI_AuthUser & Chr(39) & ") AND (" & MPTTable &
    ".Tasknumber=" & Val(Tasknum) & ");", dbReadOnly)
    Tpscbol = False
    If TPSCData.EOF = True And TPSCData.BOF = True Then Tpscbol = True
    Set TPSCData = TPSCdb.OpenRecordset(MPTTable, dbReadOnly)
    Set RUdata = RUdb.OpenRecordset("SELECT DISTINCTROW RUS.USERID, RUS.Session, RUS.MTP, RUS.MTP_Task, RUS.RU, RUS.Update, RUS.RU_Desc,
    RUS.Number From RUS WHERE ((RUS.USERID Like " & Chr(39) & CGI_AuthUser & Chr(39) & ") AND (RUS.MTP Like " & Chr(39) & MPTTable &
    Chr(39) & ") ORDER BY RUS.MTP_Task;", dbReadOnly)
    rudatabool = False
    If RUdata.EOF = True And RUdata.BOF = True Then rudatabool = True
    Table.FindFirst "[Counter] =" & Tasknum
    tempfile = CGI_AuthUser & "_" & Session & "_" & Tasknum & ".htm"
    temptask = Table.Fields(3)
    Pos = Instr(1, temptask, "(", 1)
    If Pos = 0 Then Pos = Len(Table.Fields(3)) + 1
    temptask = Left$(temptask, Pos - 1)
    Send ("Content-type: text/html")
    Send ("")
    Send ("<HTML><HEAD><TITLE> Rate Training Audience Task - " & temptask & " </TITLE>")
    Send ("<BODY bgcolor=""#00FFFF"" text=""#000000"">")
    Send ("<H2><Center> Rate Steps and Performance Measures for " & Mtpname & " performing " & temptask & "</Center></H2></HEAD>")

```

```

Send ("<FORM METHOD= "POST" ACTION= "/win.cgi/makeru.exe">")
Send ("<INPUT TYPE= "HIDDEN" NAME= "SESSION" VALUE= "" & Session & ""> ")
Send ("<INPUT TYPE= "HIDDEN" NAME= "FILE" VALUE= "" & tempfile & ""> ")
Send ("<INPUT TYPE= "HIDDEN" NAME= "MTPTABLE" VALUE= "" & MTPTable & ""> ")
Send ("<INPUT TYPE= "HIDDEN" NAME= "MTPNAME" VALUE= "" & Mtpname & ""> ")
Send ("<INPUT TYPE= "HIDDEN" NAME= "TASK" VALUE= "" & Table.Fields(6) & ""> ")
Send ("<br>")
Send ("<IMAGE SRC= "/input/images/rain_lin.gif" width=100%>")
Send ("<table border=1>")
If Table.Fields(2) <> "" Then Send ("<td><font size=+1><B>TASK NUMBER: </B></font>" + Table.Fields(2) + "<br>")
If Table.Fields(1) <> "" Then Send ("<font size=+1><B>ELEMENT: </B></font>" + Table.Fields(1) + "<br>")
If Table.Fields(3) <> "" Then Send ("<font size=+1><B>TASK: </B></font>" + Table.Fields(3) + "<br>")
If Table.Fields(4) <> "" Then Send ("<font size=+1><B>CONDITIONS: </B></font>" + Table.Fields(4) + "<br>")
If Table.Fields(5) <> "" Then Send ("<font size=+1><B>STANDARDS: </B></font>" + Table.Fields(5) + "<br>")
Send ("<p>")
Send ("<td><td align=left valign=bottom><b>TPSC TASK STEP/ PERFORMANCE MEASURE RATINGS</p><p> Use the Blocks below to rate how well
the simulation will support the individual Steps and Performance Measures of this Task.<p>")
Send ("<U><b>HIGHLY SUPPORTED (HIGH):</b></U>The TS/ PM can be performed completely. Sufficient cues are present within the
simulation environment and appropriate responses are supported such that the training experience is much the same as it is in a field
environment.<br>")
Send ("<U><b>MODERATELY SUPPORTED (MODERATE):</b></U><sup>*</sup>Most of the TS/ PM is supported. Sufficient cues are present within
the simulation environment and appropriate responses are supported to permit practice of Command/ Staff procedures and techniques;
However, missing cues/ responses will not cause negative training.<br>")
Send ("<U><b>NOT REQUIRED:</b></U><sup>*</sup>This TS/ PM does not need to be supported in the simulation environment.<p>")
Send ("<sup>*</sup></sup><i>Indicates that comments are required.</i></B></td><td align=center valign=bottom><b> Use the comment fields
below to describe the Information Products and Delivery Means required to support the step/ measure.<p> Use the following format:
<i>Product=Means</i><p>Example: Overlay=MCS,Estimate=Fax</B></td></tr>")
TaskNum = Table.Fields(6)
Table.Close
Table.FindFirst "[MPTaskReference] =" & TaskNum
Do Until Table.EOF
Table.MoveNext
If Not Table.EOF Then
If Table.Fields(4) = TaskNum Then
Nextrec = Table.Fields(2)
Else
Nextrec = 0
End If
Else
Nextrec = 0
End If
Table.MovePrevious
Currentfield2 = Table.Fields(2)
Currentfield4 = Table.Fields(4)
Currentfield5 = Table.Fields(5)
Currentfield6 = Table.Fields(6)
If Table.Fields(4) = TaskNum Then
If Currentfield2 = 4 Then Send ("<TD><B>" + Table.Fields(3) + "</B>")
If Currentfield2 = 5 Then Send ("<TD><B>" + Chr(187) + Table.Fields(3) + "</B>")
If Currentfield2 = 6 Then Send ("<TD><B>" + Chr(187) + Chr(187) + Table.Fields(3) + "</B>")

```

[illegible]

```

        Send ("<td>Products=Means<br><input type=Text name ="name-" & Currentfield4 & "-" & Currentfield5 & "-" & Currentfield6 & "
        ""value=" & Prodpipe & "" Size = 20 >")
    End If
    End If
    Send ("</td></tr>")
    End If

    If Table1.Fields(4) <> Tasknum Then GoTo bob
    Table1.MoveNext
    Loop
bob:
    Send ("</table>")
    TPSCData.Close

    Set TPSCData = TPSCdb.OpenRecordset("SELECT DISTINCTROW PTA_Xs.UserId, PTA_Xs.Session, PTA_Xs.MTP, PTA_Xs.Tasknumber, PTA_Xs.Update,
    PTA_Xs.Rating, PTA_Xs.COMMENTS From PTA_Xs WHERE ((PTA_Xs.UserId Like " & Chr(39) & CGI_AuthUser & Chr(39) & ") AND (PTA_Xs.MTP Like "
    & Chr(39) & MTPTable & Chr(39) & "))ORDER BY PTA_Xs.Tasknumber;")
    PTASBOL = False
    If TPSCData.EOF = True And TPSCData.EOF = True Then PTASBOL = True
    Comments = ""
    If PTASBOL = False Then
        TPSCData.MoveFirst
    Do Until TPSCData.EOF
        If TPSCData.Fields(6) <> "" And TPSCData.Fields(0) = CGI_AuthUser And TPSCData.Fields(2) = MTPTable And TPSCData.Fields(3) = Tasknum
        Then Comments = TPSCData.Fields(6)
        TPSCData.MoveNext
    Loop
    End If
    Send ("</SELECT></center>")
    Send ("<p><center><font size =+1>Provide any additional information you feel necessary to conduct this task.</font></center><p>")
    Send ("<center><TEXTAREA NAME="COMMENTS" Rows=5 Cols=50>" & Comments & "</Textarea></center><p>")

    Table1.Close
    Set Table1 = Db.OpenRecordset("WTPDIR", dbReadonly)

    Send ("<p><center><font size=+1>Select Represented Units for this Task</font></center><p>")
    Send ("<Center><SELECT NAME="RU" MULTIPLE SIZE=6>")
    Send ("<option value="NONE">NONE")
    ' xx = FreeFile
    ' Open "c:\netscape\server\docs\input\temp\test.txt" For Output As #xx
    ' Close #xx

    Table1.MoveFirst
    Do Until Table1.EOF
        RUBool = False

        If rudatabool = False Then

```

```

RUDATA.MoveFirst
Do Until RUDATA.EOF
    If RUDATA.Fields(0) = CGI_AuthUser And RUDATA.Fields(4) = Left(Table1.Fields(1), 8) And RUDATA.Fields(3) = Tasknum And
RUDATA.Fields(2) = MTPTable Then RUBool = True
    RUDATA.MoveNext
Loop
End If
If RUBool = True Then
    Send ("<OPTION SELECTED value = "" & Left(Table1.Fields(1), 8) & "">" & Table1.Fields(8))
Else
    Send ("<OPTION value = "" & Left(Table1.Fields(1), 8) & "">" & Table1.Fields(8))
End If

Table1.MoveNext
Loop
Send ("</select>")

Send ("<p><center><INPUT TYPE=""SUBMIT"" VALUE= ""Submit Ratings"" ><input type=""Reset"" Value=""Clear Current
Changes""></center>")

Send ("</FORM>")
Send ("</BODY>")
Send ("</HTML>")
Table1.Close
TPSCData.Close
TPSCdb.Close
Db1.Close
Db.Close
RUDATA.Close
RUDb.Close
End Sub

```

# APPENDIX F - SOURCE CODE FOR DISPLAY AND SELECTION OF REPRESENTED UNIT TASKS (RUXS.EXE)

```

Sub CGI_Main()
    Dim Gotit, Ruupdate, Listbool, Rutaskupdate, RUXBOOL, Ruxtabbool, TSSCbool, TSSCbool2, TSSCbool3, RUObool As Boolean
    Dim Pmcount(4), Stepcount(4), pmcounter, Stepcounter, Pmnumber, Stepnumber As Integer
    Dim Printcounter, Task, Step, PM, Task1, Step1, PM1, length, Rutasknum, RUCounterYes, RUCounterNo, TSSCode As Integer
    Dim X, a2value, i, k, l, m, n, o, p, Y, xx, Session, Lasttask, Pos, MTPTaskRefnum, j, RutaskRefnum, PTATasknum As Integer
    Dim mptext, Listtype, MTPNAME, Preference, Mptable, PTAFile, TempTask, lastlisttask, Filename, Comments, Nowthen, Rutable As String
    Dim a2PMcounter, a2Stepcounter, a2Stepcount(4) As Integer
    Dim tempdb, Db, Db1, RUdb, Userdb, Tsscdb, PTAXdb As Database
    Dim PTAXtab, PTA_Table, Table, Table1, table2, Usertable, RUQuery, RUXsQuery, RUXTAB, TSSC, TSSCQ As Recordset
    Dim Myfilename, Field3, Field4 As String
    Dim Mytabledef2, Mytabledef5 As TableDef
    Dim a2pmcount(4), a2TPSCode, a2PMCode As Double
    Dim Tempws As Workspace
    Dim PTABool As Boolean

    PTABool = False
    Ruupdate = False
    RUXBOOL = False
    Nowthen = Now
    Rutaskupdate = False
    Send ("Content-type: text/html")
    Send ("")

    Do While i < CGI_NumFormTuples
        If LCase(CGI_FormTuples(i).key) = "mptable" Then Mptable = Left(CGI_FormTuples(i).value, 8)
        If CGI_FormTuples(i).key = "SESSION" Then Session = Val(CGI_FormTuples(i).value)
        If LCase(CGI_FormTuples(i).key) = "mtpname" Then MTPNAME = CGI_FormTuples(i).value
        If CGI_FormTuples(i).key = "FILE" Then Filename = CGI_FormTuples(i).value
        If CGI_FormTuples(i).key = "COMMENTS" Then Comments = CGI_FormTuples(i).value
        If LCase(CGI_FormTuples(i).key) = "rutable" Then
            Ruupdate = True
            Rutable = CGI_FormTuples(i).value
        End If
        If CGI_FormTuples(i).key = "RUTASKSBOOL" Then Rutaskupdate = True
        If CGI_FormTuples(i).key = "RUTASKNUM" Then Rutasknum = CGI_FormTuples(i).value
        If CGI_FormTuples(i).key = "PTATASKNUM" Then
            PTATasknum = CGI_FormTuples(i).value
        End If
        i = i + 1
    Loop
    mptext = Mptable
    Send ("<HTML><CENTER><TITLE>Training Audience - " & MTPNAME & " </TITLE></CENTER>")
    Send ("<BODY TEXT=""#000000"" bgcolor=""#00FFFF"" LINK=""#0000EE"" VLINK=""#551A8B"" ALINK=""#FF0000">")

```

```

Send ("<H2><Center> WARSIM 2000 Subject Matter Expert Data Collection System <br> Represented Unit Task Selection and Evaluation
<br> Training Audience - " & MTPNAME & "</Center></H2></HEAD>")
Send ("<hr><p><B><Center><h2>Navigation:</h2></Center></B>")

Send ("<p><center><table border=1><tr><td align= center><h3><a href=/input/primary.htm>Select Another Training
Audience</a></td></tr></table></center>")
Send ("<form method="Post" action="/wincgi/mapping.exe?0,0">")
Send ("<INPUT TYPE="HIDDEN" NAME="SESSION" VALUE="" & Session & "">")
Send ("<INPUT TYPE="HIDDEN" NAME="MTPSELECT" VALUE="" & MtpTable & "">")
Send ("<INPUT TYPE="HIDDEN" NAME="FILE" VALUE="" & Filename & "">")
Send ("<P><Center><INPUT TYPE="SUBMIT" VALUE= "Select/ Deselect Training Audience Tasks">")
Send ("</form>")

Send ("<form method="Post" action="/wincgi/printit.exe">")
Send ("<INPUT TYPE="HIDDEN" NAME="SESSION" VALUE="" & Session & "">")
Send ("<INPUT TYPE="HIDDEN" NAME="MTPTABLE" VALUE="" & MtpTable & "">")
Send ("<INPUT TYPE="HIDDEN" NAME="MTPNAME" VALUE="" & MTPNAME & "">")
Send ("<INPUT TYPE="HIDDEN" NAME="FILE" VALUE="" & Filename & "">")
Send ("<INPUT TYPE="HIDDEN" NAME="first" VALUE="time">")
Send ("<P><INPUT TYPE="SUBMIT" VALUE= "Reevaluate Selected Training Audience Tasks"></center>")

Send ("</form>")
Send ("<p>")

Dbfilename = "c:\netscape\server\docs\input\temp\" & CGI_AuthUser & Month(Time) & Day(Time) & Hour(Time) & Minute(Time) &
Second(Time) & "tempdb.mdb"

' Set Tempws = DBEngine.Workspaces(0)
'
' Set tempdb = Tempws.CreateDatabase(Dbfilename, dbLangGeneral)
'
' Set Mytabledef2 = tempdb.CreateTableDef("PTA_Xs")
' Mytabledef2.Connect = ";DATABASE=c:\netscape\server\docs\input\database\mtptpsc.mdb"
' Mytabledef2.SourceTableName = "PTA_Xs"
' tempdb.TableDefs.Append Mytabledef2
'
' Set Mytabledef5 = tempdb.CreateTableDef("RU_Xs")
' Mytabledef5.Connect = ";DATABASE=c:\netscape\server\docs\input\database\mtpru.mdb"
' Mytabledef5.SourceTableName = "RU_Xs"
' tempdb.TableDefs.Append Mytabledef5

Set RUdb = OpenDatabase("c:\netscape\server\docs\input\database\mtpru.mdb")
Ruxtabbool = False

***** Working here *****

```

```

' "SELECT DISTINCTROW " & Rutable & ".RU_Task, " & Rutable & ".RU_Step, " & Rutable & ".RU_PM, " & Rutable & ".Rating, " & Rutable &
'.USERID) AND (RU_Xs.MTP = " & Rutable & ".MTP) AND (RU_Xs.MTP_Task = " & Rutable & ".MTP_Task) AND (RU_Xs.MTP_Task = " & Rutable &
".RU_Task)) ON (RU_Xs.USERID = RU_Xs.USERID) AND (RU_Xs.MTP = RU_Xs.MTP) AND (RU_Xs.MTP_Task = RU_Xs.MTP_Task) AND (RU_Xs.MTP_Task = RU_Xs.MTP_Task) ON
(PTA_Xs.UserID = RU_Xs.USERID) AND (PTA_Xs.MTP = RU_Xs.MTP) AND (PTA_Xs.TaskNumber = RU_Xs.TaskNumber) ON Data.Login = PTA_Xs.UserID WHERE
((PTA_Xs.Update Like " & Chr(39) & "yes" & Chr(39) & ") AND (Data.Display User?)=Yes) AND (" & Userstring & ") AND (PTA_Xs.MTP Like "
& Chr(39) & Mtpable & Chr(39) & ") AND (RU_Xs.RU Like " & Chr(39) & Rutable & Chr(39) & _
& ") AND (RU_Xs.RU_Task=" & Val(Rutasknum) & ") AND (PTA_Xs.TaskNumber= " & Val(Tasknum) & "))) ORDER BY " & Rutable & ".RU_Task, " &
Rutable & ".RU_Step, " & Rutable & ".RU_PM, " & Rutable & ".RU_Task, " &
' Set RUXTAB = tempdb.OpenRecordset("SELECT DISTINCTROW RU_Xs.USERID, RU_Xs.Session, RU_Xs.MTP, RU_Xs.MTP_Task, RU_Xs.RU,
RU_Xs.RU_Task, RU_Xs.Update, RU_Xs.TSSC, RU_Xs.Comments, RU_Xs.RU_Task_Desc From RU_Xs, PTA_Xs WHERE ((PTA_Xs.Update Like " & Chr(39) &
"yes" & Chr(39) & ") AND (PTA_Xs.MTP Like " & Chr(39) & Mtpable & Chr(39) & ") AND (PTA_Xs.TaskNumber = " & PTATasknum & ") AND
(RU_Xs.USERID Like " & Chr(39) & CGI_AuthUser & Chr(39) & ") AND (RU_Xs.MTP Like " & Chr(39) & Mtpable & Chr(39) & ") AND (RU_Xs.RU
Like " & Chr(39) & Rutable & Chr(39) & ") ORDER BY RU_Xs.MTP_Task, RU_Xs.RU_Task, RU_Xs.Update;")
' (PTA_Xs.Update Like " & Chr(39) & "yes" & Chr(39) & ") AND (PTA_Xs.MTP Like " & Chr(39) & Mtpable & Chr(39) & ") AND
(PTA_Xs.TaskNumber = " & PTATasknum & ") AND
Set RUXTAB = RUdb.OpenRecordset("SELECT DISTINCTROW RU_Xs.USERID, RU_Xs.Session, RU_Xs.MTP, RU_Xs.MTP_Task, RU_Xs.RU, RU_Xs.RU_Task,
RU_Xs.Update, RU_Xs.TSSC, RU_Xs.Comments, RU_Xs.RU_Task_Desc From RU_Xs WHERE ((RU_Xs.USERID Like " & Chr(39) & CGI_AuthUser & Chr(39)
& ") AND (RU_Xs.MTP Like " & Chr(39) & Mtpable & Chr(39) & ") AND (RU_Xs.RU Like " & Chr(39) & Rutable & Chr(39) & "))) ORDER BY
RU_Xs.MTP_Task, RU_Xs.RU_Task, RU_Xs.Update;")
If RUXTAB.EOF = True And RUXTAB.EOF = True Then Ruxtabbool = True
RUXTAB.LockEdits = False

' Set RUXTAB = RUdb.OpenRecordset("RU_Xs")

Set Db = OpenDatabase("c:\netscape\server\docs\input\database\mtptask.mdb")

' next block updates ru_xs

If Ruupdate = True And Rutaskupdate = False Then

Set Table = Db.OpenRecordset(Rutable)

i = 0
Do While i < CGI_NumFormTuples
If Left(CGI_FormTuples(i).key, 8) = "ruselect" Then

Pos = InStr(1, CGI_FormTuples(i).value, "_", 1)
If Pos = 0 Then Pos = Len(Table.Fields(3)) + 1
MTPTaskRefnum = Val(Left$(CGI_FormTuples(i).value, Pos - 1))
j = Len(CGI_FormTuples(i).value) - Pos
RUTaskRefnum = Val(Right$(CGI_FormTuples(i).value, j))

If Ruxtabbool = False Then
RUXTAB.MoveFirst

```

```

RUXBOOL = False
Do Until RUXTAB.EOF
    If RUXTAB.Fields(0) = CGI_AuthUser And RUXTAB.Fields(2) = Mtptable And RUXTAB.Fields(3) = MTPTaskRefnum And RUXTAB.Fields(4) =
    Rutable And RUXTAB.Fields(5) = RUTaskRefNUM And RUXTAB.Fields(6) = "Yes" Then
        RUXBOOL = True
        RUXTAB.Edit
        RUXTAB.Fields(6) = "Save_Yes"
        RUXTAB.Update
    ElseIf RUXTAB.Fields(0) = CGI_AuthUser And RUXTAB.Fields(2) = Mtptable And RUXTAB.Fields(3) = MTPTaskRefnum And
    RUXTAB.Fields(4) = Rutable And RUXTAB.Fields(5) = RUTaskRefNUM And RUXTAB.Fields(6) = "No" Then
        RUXBOOL = True
        RUXTAB.Edit
        RUXTAB.Fields(6) = "Save_No"
        RUXTAB.Update
    End If
    RUXTAB.MoveNext
Loop
End If

If RUXBOOL = False Then
    RUXTAB.AddNew
    RUXTAB.Fields(0) = CGI_AuthUser
    RUXTAB.Fields(1) = Session
    RUXTAB.Fields(2) = Mtptable
    RUXTAB.Fields(3) = MTPTaskRefnum
    RUXTAB.Fields(4) = Rutable
    RUXTAB.Fields(5) = RUTaskRefNUM
    RUXTAB.Fields(6) = "Save_No"
    RUXTAB.Fields(7) = 0
    RUXTAB.Fields(8) = " "
    Table.MoveFirst
Do Until Table.EOF
    If Table.Fields(6) = RUTaskRefNUM Then
        Temptask = Table.Fields(3)
        Pos = InStr(1, Temptask, ",", 1)
        If Pos = 0 Then Pos = Len(Table.Fields(3)) + 1
        Temptask = Left$(Temptask, Pos - 1)
        RUXTAB.Fields(9) = Temptask
        Table.MoveLast
        RUXTAB.Update
    End If
    Table.MoveNext
Loop
End If

End If

```

```

i = i + 1
Loop

Ruxtabbool = False
If RUXTAB.EOF = True And RUXTAB.EOF = True Then Ruxtabbool = True

If Ruxtabbool = False Then
RUXTAB.MoveFirst
Do Until RUXTAB.EOF

    If RUXTAB.Fields(0) = CGI_AuthUser And RUXTAB.Fields(2) = Mtptable And RUXTAB.Fields(4) = Rutable And RUXTAB.Fields(3) =
MTPTASKRefnum And (RUXTAB.Fields(6) = "No" Or RUXTAB.Fields(6) = "Save_Yes") Then
        RUXTAB.Delete
    ElseIf RUXTAB.Fields(0) = CGI_AuthUser And RUXTAB.Fields(2) = Mtptable And RUXTAB.Fields(4) = Rutable And RUXTAB.Fields(3) =
MTPTASKRefnum And RUXTAB.Fields(6) = "Save_Yes" Then
        RUXTAB.Edit
        RUXTAB.Fields(6) = "Yes"
        RUXTAB.Update
    ElseIf RUXTAB.Fields(0) = CGI_AuthUser And RUXTAB.Fields(2) = Mtptable And RUXTAB.Fields(4) = Rutable And RUXTAB.Fields(3) =
MTPTASKRefnum And RUXTAB.Fields(6) = "Save_No" Then
        RUXTAB.Edit
        RUXTAB.Fields(6) = "No"
        RUXTAB.Update
    End If

End If

RUXTAB.MoveNext
Loop
End If
Table.Close
RUXTAB.Close

End If
'
' end of ru_x update
'
'
' xx = FreeFile
' Open "c:\netscape\server\docs\input\temp\test.txt" For Output As #xx
' Print #xx, Mtptable
' Close #xx

```

```

If Ruupdate = True And Rutaskupdate = True Then

    Set Tsscdb = OpenDatabase("c:\netscape\server\docs\input\database\mtpru.mdb")

    '
    ' Set TSSC = Tsscdb.OpenRecordset(Rutable)
    Set TSSC = Tsscdb.OpenRecordset("SELECT DISTINCTROW " & Rutable & ".USERID, " & Rutable & ".Session, " & Rutable & ".MTP, " &
    Rutable & ".MTP_Task, " & Rutable & ".RU_Task, " & Rutable & ".RU_STEP, " & Rutable & ".RU_PM, " & Rutable & ".Rating, " & Rutable &
    ".Equip, " & Rutable & ".Object, " & Rutable & ".Comment From " & Rutable & " WHERE ((" & Rutable & ".USERID Like " & Chr(39) &
    CGI_AuthUser & Chr(39) & ") AND (" & Rutable & ".MTP Like " & Chr(39) & Mtpable & Chr(39) & ") AND (" & Rutable & ".MTP_Task=" &
    Val(PTATasknum) & ") AND (" & Rutable & ".RU_Task=" & Rutasknum & "));")

    TSSCBOol = False
    If TSSC.BOF = True And TSSC.EOF = True Then TSSCBOol = True
    TSSC.Lockedits = False

    If TSSCBOol = False Then
        TSSC.MoveFirst

    Do Until TSSC.EOF
        If TSSC.Fields(0) = CGI_AuthUser And TSSC.Fields(2) = Mtpable And TSSC.Fields(3) = PTATasknum And Val(TSSC.Fields(4)) =
        Rutasknum Then TSSC.Delete
        TSSC.MoveNext
    Loop

    End If

i = 0
Do While i < CGI_NumFormTuples

    If Left$(CGI_FormTuples(i).key, 4) = "tssc" Then

        TSSC.AddNew
        j = 6
        k = j
        l = 2
        length = Len(CGI_FormTuples(i).key)
        Do While j < length + 1
            If Mid(CGI_FormTuples(i).key, j, 1) = "-" Or Mid(CGI_FormTuples(i).key, j, 1) = "+" Then
                If l = 2 Then Task = Val(Mid(CGI_FormTuples(i).key, k, j - k))
                If l = 3 Then Step = Val(Mid(CGI_FormTuples(i).key, k, j - k))
                If l = 4 Then PM = Val(Mid(CGI_FormTuples(i).key, k, j - k))
                l = l + 1
                k = j + 1
            End If
            j = j + 1
        Loop
    End If

    i = i + 1

```

```

m = 0
Do While m < CGI_NumFormTuples
  If Left$(CGI_FormTuples(m).key, 4) = "equip" Then
    j = 6
    k = j
    l = 2
    Task1 = -1
    Step1 = -1
    PM1 = -1
    length = Len(CGI_FormTuples(i).key)
    Do While j < length + 1
      If Mid(CGI_FormTuples(m).key, j, 1) = "-" Or Mid(CGI_FormTuples(m).key, j, 1) = "+" Then
        If l = 2 Then Task1 = Val(Mid(CGI_FormTuples(m).key, k, j - k))
        If l = 3 Then Step1 = Val(Mid(CGI_FormTuples(m).key, k, j - k))
        If l = 4 Then PM1 = Val(Mid(CGI_FormTuples(m).key, k, j - k))
        If Task1 = Task And Task1 >= 0 And Step1 >= 0 And PM1 >= 0 And Step1 = Step And PM1 = PM And CGI_FormTuples(m).value <>
          "" Then
            TSSC.Fields(8) = CGI_FormTuples(m).value
            End If
            l = l + 1
            k = j + 1
            End If
            j = j + 1
            Loop
          End If
          m = m + 1
        Loop
      
```

```

m = 0
Do While m < CGI_NumFormTuples
  If Left$(CGI_FormTuples(m).key, 4) = "objs" Then
    j = 6
    k = j
    l = 2
    Task1 = -1
    Step1 = -1
    PM1 = -1
    length = Len(CGI_FormTuples(i).key)
    Do While j < length + 1
      If Mid(CGI_FormTuples(m).key, j, 1) = "-" Or Mid(CGI_FormTuples(m).key, j, 1) = "+" Then
        If l = 2 Then Task1 = Val(Mid(CGI_FormTuples(m).key, k, j - k))
        If l = 3 Then Step1 = Val(Mid(CGI_FormTuples(m).key, k, j - k))
        If l = 4 Then PM1 = Val(Mid(CGI_FormTuples(m).key, k, j - k))
        If Task1 = Task And Task1 >= 0 And Step1 >= 0 And PM1 >= 0 And Step1 = Step And PM1 = PM And CGI_FormTuples(m).value <>
          "" Then
            TSSC.Fields(9) = CGI_FormTuples(m).value
            End If
            l = l + 1
            k = j + 1
            End If
            j = j + 1
            Loop
          End If
          m = m + 1
        Loop
      
```

```

End If
    l = l + 1
    k = j + 1
End If
    j = j + 1
Loop
End If
    m = m + 1
Loop

m = 0
Do While m < CGI_NumFormTuples
    If Left$(CGI_FormTuples(m).key, 4) = "cmts" Then
        j = 6
        k = j
        l = 2
        Task1 = -1
        Step1 = -1
        PM1 = -1
        length = Len(CGI_FormTuples(i).key)
        Do While j < length + 1
            If Mid(CGI_FormTuples(m).key, j, 1) = "-" Or Mid(CGI_FormTuples(m).key, j, 1) = "+" Then
                If l = 2 Then Task1 = Val(Mid(CGI_FormTuples(m).key, k, j - k))
                If l = 3 Then Step1 = Val(Mid(CGI_FormTuples(m).key, k, j - k))
                If l = 4 Then PM1 = Val(Mid(CGI_FormTuples(m).key, k, j - k))
                If Task1 = Task And Task1 >= 0 And Step1 >= 0 And PM1 >= 0 And Step1 = Step And PM1 = PM And CGI_FormTuples(m).value <>
                    "" Then
                        TSSC.Fields(10) = CGI_FormTuples(m).value
                    End If
                    l = l + 1
                    k = j + 1
                End If
                j = j + 1
            Loop
            End If
            m = m + 1
        Loop

TSSC.Fields(0) = CGI_AuthUser
TSSC.Fields(1) = Session
TSSC.Fields(2) = Mtptable
TSSC.Fields(3) = PrtAtasknum
TSSC.Fields(4) = Task
TSSC.Fields(5) = Step
TSSC.Fields(6) = PM
TSSC.Fields(7) = CGI_FormTuples(i).value

```

```
TSSC.Update
End If
```

```
i = i + 1
Loop
```

```
TSSC.Close
```

```
, 5 sec
xx = FreerFile
Open "c:\netscape\server\docs\input\temp\testru.txt" For Output As #xx
Print #xx, CGI_AuthUser & " "; Now & " " & Nowthen
Close #xx
```

```
TSSCBool3 = False
```

```
Set TSSCQ = TsscdB.OpenRecordset("SELECT DISTINCTROW " & Rutable & ".USERID, " & Rutable & ".Session, " & Rutable & ".MTP, " & Rutable  
& ".MTP_Task, " & Rutable & ".RU_Task, " & Rutable & ".RU_PM, " & Rutable & ".Rating From " & Rutable & "  
Where (" & Rutable & ".USERID Like "" & CGI_AuthUser & """) And (" & Rutable & ".MTP Like "" & Mptable & """) And (" & Rutable & "  
".MTP_Task = " & PTATasknum & ") AND (" & Rutable & ".RU_Task = " & Rutasknum & ") ORDER BY " & Rutable & ".RU_Task DESC, " & Rutable  
& ".RU_Step DESC, " & Rutable & ".RU_PM DESC;", dbReadonly)
```

```
If TSSCQ.EOF = True And TSSCQ.EOF = True Then TSSCBool3 = True
```

```
TSSCBool2 = False
```

```
Set TSSC = TsscdB.OpenRecordset("SELECT DISTINCTROW RU_Xs.USERID, RU_Xs.Session, RU_Xs.MTP, RU_Xs.MTP_Task, RU_Xs.RU, RU_Xs.RU_Task,  
RU_Xs.Update, RU_Xs.TSSC, RU_Xs.Comments, RU_Xs.RU_Task_Desc, RU_Xs.TSSC2 From RU_Xs WHERE ((RU_Xs.USERID Like " & Chr(39) & "  
CGI_AuthUser & Chr(39) & ") AND (RU_Xs.MTP Like " & Chr(39) & Mptable & Chr(39) & ") AND (RU_Xs.RU_Task = " & Rutasknum & ") And  
(RU_Xs.RU Like " & Chr(39) & Rutable & Chr(39) & ") ORDER BY RU_Xs.MTP_Task, RU_Xs.RU_Task, RU_Xs.Update;")  
If TSSC.EOF = True And TSSC.EOF = True Then  
, Send ("No match <p>")  
TSSCBool2 = True  
End If  
TSSC.LockEdits = False
```

```
, Set TSSC = TsscdB.OpenRecordset("RU_Xs")  
TSSCCode = -1
```

```
If TSSCBool3 = False Then  
TSSCQ.MoveFirst
```

```

i = 0
Do Until i = 5
    Pmcount(i) = 0
    pmcounter = 0
    Stepcounter = 0
    Stepcount(i) = 0
    a2pmcount(i) = 0
    a2PMcounter = 0
    a2Stepcounter = 0
    i = i + 1
Loop
'

Do Until TSSCQ.EOF
    If Val(TSSCQ.Fields(4)) = Rutasknum Then
        Pmnumber = TSSCQ.Fields(6)
        Stepnumber = TSSCQ.Fields(5)
        Tasknumber = TSSCQ.Fields(4)
        If TSSCQ.Fields(7) = "high" Then
            value = 3
            a2value = 4
        ElseIf TSSCQ.Fields(7) = "med" Then
            value = 2
            a2value = 3
        ElseIf TSSCQ.Fields(7) = "low" Then
            value = 1
            a2value = 2
        ElseIf TSSCQ.Fields(7) = "none" Then
            value = 0
            a2value = 0
        End If
        If Pmnumber >= 1 Then
            Pmcount(value) = Pmcount(value) + 1
            a2pmcount(a2value) = a2pmcount(a2value) + 1
            pmcounter = pmcounter + 1
            a2PMcounter = a2PMcounter + 1
        Else ' pmnumber=0
            Stepcounter = Stepcounter + 1
            a2Stepcounter = a2Stepcounter + 1
            Stepcount(value) = Stepcount(value) + 1
            a2pmcount(a2value) = a2pmcount(a2value) + 1
        End If
        If Pmnumber = 1 Then ' compute step rating
            Stepcounter = Stepcounter + 1
            a2Stepcounter = a2Stepcounter + 1
            Send ("*** a2pmcode = " & a2PMcode & "a2pmcounter = " & a2PMcounter & "<p>")
            If a2PMcounter <> 0 Then
                a2PMcode = a2PMcode + (((a2pmcount(4) * 4) + (a2pmcount(3) * 3) + (a2pmcount(2) * 2) + (a2pmcount(1) * 2) +
(a2pmcount(0) * 0)) / a2PMcounter)

```

```

Else
    a2PMCode = a2PMCode + ((a2pmcount(4) * 4) + (a2pmcount(3) * 3) + (a2pmcount(2) * 2) + (a2pmcount(1) * 2) +
(a2pmcount(0) * 0))
End If

,
    Send ("counts4 = " & a2pmcount(4) & "<p>counts3 = " & a2pmcount(3) & "<p>counts2 = " & a2pmcount(2) & "<p>counts3 = " &
a2pmcount(1) & "<p>counts0 = " & a2pmcount(0) & "<p>a2pmcount = " & a2PMCode & "<p>a2pmcode = " & a2PMCode & "<p>")
    Send ("a2stepcounter = " & a2Stepcounter & " stepcounter = " & Stepcounter & "<p>")

a2PMcount = 0
X = 0
Do Until X = 5
    a2pmcount(X) = 0
    X = X + 1
Loop

If pmcount = 2 And Pmcount(3) = 2 Then
    Stepcount(3) = Stepcount(3) + 1
ElseIf pmcount = 2 And Pmcount(3) + Pmcount(2) = 2 Then
    Stepcount(2) = Stepcount(2) + 1
ElseIf pmcount = 2 And Pmcount(3) + Pmcount(2) = 1 And Pmcount(1) = 1 Then
    Stepcount(1) = Stepcount(1) + 1
ElseIf pmcount = 2 And Pmcount(1) + Pmcount(0) > 0 Then
    Stepcount(0) = Stepcount(0) + 1
ElseIf pmcount >= 3 And (Pmcount(3) / pmcount) >= 0.66 And Pmcount(0) + Pmcount(1) = 0 Then
    Stepcount(3) = Stepcount(3) + 1
ElseIf pmcount >= 3 And (Pmcount(3) + Pmcount(2)) / pmcount >= 0.66 Then
    Stepcount(2) = Stepcount(2) + 1
ElseIf pmcount >= 3 And Pmcount(1) / pmcount >= 0.25 Then
    Stepcount(1) = Stepcount(1) + 1
Else
    Stepcount(0) = Stepcount(0) + 1
End If

ElseIf Pmnumber = 0 Then
    a2PMCode = a2PMCode + ((a2pmcount(4) * 4) + (a2pmcount(3) * 3) + (a2pmcount(2) * 2) + (a2pmcount(1) * 2) + (a2pmcount(0) *
0)))

,
    Send ("zero ..... counts4 = " & a2pmcount(4) & "<p>counts3 = " & a2pmcount(3) & "<p>counts2 = " & a2pmcount(2) &
"<p>counts1 = " & a2pmcount(1) & "<p>counts0 = " & a2pmcount(0) & "<p>a2pmcount = " & a2PMCode & "<p>a2pmcode = " & a2PMCode &
"<p>")
    Send ("a2stepcounter = " & a2Stepcounter & " stepcounter = " & Stepcounter & "<p>")

a2Stepcounter = 0
a2PMCode = 0
,deleted 5 feb
, added next two lines 5 feb
    Stepcounter = Stepcounter + 1
    a2Stepcounter = a2Stepcounter + 1
,
X = 0
Do Until X = 5
    a2pmcount(X) = 0

```

```

X = X + 1
Loop

End If

If (Pmnumber = 1 Or Pmnumber = 0) And Stepnumber = 1 Then
  If Stepcounter <> 0 Then
    a2TPSCode = a2PMCode / Stepcounter
  Else
    a2TPSCode = a2PMCode
  End If

  If Stepcounter = 1 And Stepcount(3) = 1 Then
    TSSCode = 4
  ElseIf Stepcounter = 1 And Stepcount(2) = 1 Then
    TSSCode = 3
  ElseIf Stepcounter = 2 And Stepcount(3) = 2 Then
    TSSCode = 4
  ElseIf Stepcounter = 2 And (Stepcount(3) + Stepcount(2)) = 2 Then
    TSSCode = 3
  ElseIf Stepcounter = 2 And (Stepcount(3) + Stepcount(2)) = 1 And Stepcount(1) = 1 Then
    TSSCode = 2
  ElseIf Stepcounter = 2 And Stepcount(1) >= 1 Then
    TSSCode = 1
  ElseIf Stepcounter = 2 Then
    TSSCode = 0
  ElseIf Stepcounter >= 3 And Stepcount(3) = Stepcounter Then
    TSSCode = 4
  ElseIf Stepcounter >= 3 And (Stepcount(3) / Stepcounter) >= 0.66 And (Stepcount(0) + Stepcount(1)) / Stepcounter <= 0.25
    Then
      TSSCode = 3
    ElseIf Stepcounter >= 3 And (Stepcount(3) + Stepcount(2)) / Stepcounter >= 0.66 Then
      TSSCode = 2
    ElseIf Stepcounter >= 3 And (Stepcount(3) + Stepcount(2)) / Stepcounter >= 0.3 Then
      TSSCode = 1
    Else
      TSSCode = 0
    End If

    i = 0
    Do Until i = 5
      Pmcount(i) = 0
      a2pmcount(i) = 0
      Stepcount(i) = 0
      i = i + 1
    Loop
    a2PMCode = 0
    pmcount = 0
    Stepcounter = 0

```

```

End If
End If

' TSSC.MoveFirst

' Do Until TSSC.EOF
' If Val(TSSC.Fields(5)) = Rutasknum And TSSC.Fields(4) = Rutable And Val(TSSC.Fields(3)) = PTATasknum And TSSC.Fields(2) = Mtptable
And TSSC.Fields(0) = CGI_AuthUser Then
' TSSC.Edit
' TSSC.Fields(6) = "Yes"
' If Comments <> "" Then TSSC.Fields(8) = Comments
' If Comments = "" Then TSSC.Fields(8) = " "
' TSSC.Fields(7) = TSSCCode
' Send (TSSCCode & "<p>")
' TSSC.Update
' End If
' TSSC.MoveNext
' Loop

TSSCQ.MoveNext
Loop

End If

TSSC.MoveFirst

Do Until TSSC.EOF
If Val(TSSC.Fields(5)) = Rutasknum And TSSC.Fields(4) = Rutable And Val(TSSC.Fields(3)) = PTATasknum And TSSC.Fields(2) = Mtptable
And TSSC.Fields(0) = CGI_AuthUser Then
TSSC.Edit
TSSC.Fields(6) = "Yes"
If Comments <> "" Then TSSC.Fields(8) = Comments
If Comments = "" Then TSSC.Fields(8) = " "
TSSC.Fields(7) = TSSCCode
TSSC.Fields(10) = a2TPSCode
' Send (TSSCCode & "<p>")
TSSC.Update
End If
TSSC.MoveNext
Loop

```

```
TSSCQ.Close
TSSC.Close
Tsscdb.Close
End If
```

```
xx = FreeFile
Open "c:\netscape\server\docs\input\temp\testru2.txt" For Output As #xx
Print #xx, CGI_AuthUser & " "; Now & " " & NowThen
Close #xx

'
' tasks update
'

Set Tsscdb = OpenDatabase("c:\netscape\server\docs\input\database\mtpru.mdb")
```

```
TSSCB00L2 = False
Set TSSC = TSSCdb.OpenRecordset("SELECT DISTINCTROW RU_Xs.USERID, RU_Xs.Session, RU_Xs.MTP, RU_Xs.MTP_Task, RU_Xs.RU, RU_Xs.RU_Task, RU_Xs.Update, RU_Xs.TSSC, RU_Xs.Comments, RU_Xs.RU_Task_Desc From RU_Xs WHERE ((RU_Xs.USERID Like " & Chr(39) & CGI_UserName & Chr(39) & ") AND (RU_Xs.MTP Like " & Chr(39) & MtpTable & Chr(39) & ") AND (RU_Xs.RU Like " & Chr(39) & Rutable & Chr(39) & "))ORDER BY RU_Xs.MTP_Task, RU_Xs.RU_Task, RU_Xs.Update:")
If TSSC.EOF = True And TSSC.EOF = True Then TSSCB00L2 = True
TSSC.LockEdits = False
```

```

' Set TSSC = TsscdB.OpenRecordset ("RU_Xs")

    RUCounterYes = 0
    RUCounterNo = 0

    If TSSCBOOL2 = False Then
        TSSC.MoveFirst
    End If

```

```

Do Until TSSC.EOF
    If TSSC.Fields(0) = CGI_AuthUser And TSSC.Fields(2) = Mtptable And Val(TSSC.Fields(3)) = PTATasknum And TSSC.Fields(4) = Rutable
    And TSSC.Fields(6) = "Yes" Then
        RUCounterYes = RUCounterYes + 1
    End If
    If TSSC.Fields(0) = CGI_AuthUser And TSSC.Fields(2) = Mtptable And Val(TSSC.Fields(3)) = PTATasknum And TSSC.Fields(4) = Rutable
    And TSSC.Fields(6) = "No" Then
        RUCounterNo = RUCounterNo + 1
    End If
    TSSC.MoveNext
End Until

```

```

Loop
End If
TSSC.Close

TSSCBOOL2 = False
Set TSSC = TsscdB.OpenRecordset("SELECT DISTINCTROW RUS.USERID, RUS.Session, RUS.MTP, RUS.MTP_Task, RUS.RU, RUS.Update, RUS.RU_Desc,
RUS.Number From RUS WHERE (RUS.USERID Like " & Chr(39) & CGI_AuthUser & Chr(39) & ") AND (RUS.MTP Like " & Chr(39) & Mtptable &
Chr(39) & ") ORDER BY RUS.MTP_Task;")
If TSSC.EOF = True And TSSC.EOF = True Then TSSCBOOL2 = True
TSSC.LockEdits = False

' Set TSSC = TsscdB.OpenRecordset("RUS")
If TSSCBOOL2 = False Then
    TSSC.MoveFirst
    Do Until TSSC.EOF
        If TSSC.Fields(0) = CGI_AuthUser And TSSC.Fields(2) = Mtptable And Val(TSSC.Fields(3)) = PTATasknum And TSSC.Fields(4) = Rutable
        Then
            TSSC.Edit
            If RUCounterYes = 0 Then
                TSSC.Fields(5) = "No"
            ElseIf RUCounterNo > 0 Then
                TSSC.Fields(5) = "Some"
            ElseIf RUCounterNo = 0 Then
                TSSC.Fields(5) = "Yes"
            End If
            TSSC.Update
            TSSC.MoveLast
            TSSC.MovePrevious
        End If
    Loop
End If

TSSC.MoveNext
Loop
End If
TSSC.Close

TsscdB.Close

Set PTAXdb = OpenDatabase("c:\netscape\server\docs\input\database\mtptpsc.mdb")
Set Table = Db.OpenRecordset(Mtptable, dbOpenSnapshot)

```



```

If Pos = 0 Then Pos = Len(Table.Fields(3)) + 1
Temptask = Left$(Temptask, Pos - 1) & " " & Table.Fields(2)
If Lasttask > 0 And Lasttask <> RUQuery.Fields(3) Then Send ("</ul></ul>")
Send ("<h3><hr><ul><li>Training Audience Task - " & Temptask & "<ul></h3>")
End If
Send ("<form method="post" action="/wincgi/mapru.exe">")
Send ("<INPUT TYPE="HIDDEN" NAME="mtptselect" VALUE="" & Mtptable & "">")
Send ("<INPUT TYPE="HIDDEN" NAME="mtptasknum" VALUE="" & RUQuery.Fields(3) & "">")
Send ("<INPUT TYPE="HIDDEN" NAME="mtptname" VALUE="" & MTPTNAME & "">")
Send ("<INPUT TYPE="HIDDEN" NAME="ruselect" VALUE="" & RUQuery.Fields(4) & "">")
Send ("<INPUT TYPE="HIDDEN" NAME="Session" VALUE="" & Session & "">")
Send ("<INPUT TYPE="HIDDEN" NAME="PTATask" VALUE="" & Temptask & "">")
Send ("<INPUT TYPE="HIDDEN" NAME="FILE" VALUE="" & Filename & "">")
Send ("<INPUT TYPE="HIDDEN" NAME="RUNAME" VALUE="" & RUQuery.Fields(6) & "">")
Lasttask = RUQuery.Fields(3)
Send ("<li><hr><b>First,</b> Select Tasks to Evaluate for this Unit: <INPUT TYPE="SUBMIT" VALUE= "Represented Unit - " &
RUQuery.Fields(6) & "">")
Send ("</form>")

, *****
Send ("<form method="post" action="/wincgi/mapru2.exe">")
Send ("<INPUT TYPE="HIDDEN" NAME="mtptselect" VALUE="" & Mtptable & "">")
Send ("<INPUT TYPE="HIDDEN" NAME="mtptasknum" VALUE="" & RUQuery.Fields(3) & "">")
Send ("<INPUT TYPE="HIDDEN" NAME="mtptname" VALUE="" & MTPTNAME & "">")
Send ("<INPUT TYPE="HIDDEN" NAME="ruselect" VALUE="" & RUQuery.Fields(4) & "">")
Send ("<INPUT TYPE="HIDDEN" NAME="Session" VALUE="" & Session & "">")
Send ("<INPUT TYPE="HIDDEN" NAME="PTATask" VALUE="" & Temptask & "">")
Send ("<INPUT TYPE="HIDDEN" NAME="FILE" VALUE="" & Filename & "">")
Send ("<INPUT TYPE="HIDDEN" NAME="RUNAME" VALUE="" & RUQuery.Fields(6) & "">")
Lasttask = RUQuery.Fields(3)
Send ("<b>Then,</b> Evaluate These Selected Tasks for this Unit: <INPUT TYPE="SUBMIT" VALUE= "Represented Unit - " &
RUQuery.Fields(6) & "">")
Send ("</form>")

End If
End If
, Ruxtabbool = False
, Field3 = RUQuery.Fields(3)
, Field4 = RUQuery.Fields(4)
, Set RUXTAB = RUDb.OpenRecordset("SELECT DISTINCTROW RU_Xs.USERID, RU_Xs.Session, RU_Xs.MTP, RU_Xs.MTP_Task, RU_Xs.RU,
RU_Xs.RU_Task, RU_Xs.Update, RU_Xs.TSSC, RU_Xs.Comments, RU_Xs.RU_Task_Desc From RU_Xs WHERE ((RU_Xs.USERID Like " & Chr(39) &
CGI_AuthUser & Chr(39) & ") AND (RU_Xs.MTP Like " & Chr(39) & Mtptable & Chr(39) & ") AND (RU_Xs.MTP_Task= " & Val(Field3) & ") AND
(RU_Xs.RU Like " & Chr(39) & Field4 & Chr(39) & ") ORDER BY RU_Xs.RU_Task;", dbOpenSnapshot)
, If RUXTAB.EOF = True And RUXTAB.EOF = True Then Ruxtabbool = True
, If Ruxtabbool = False Then
, RUXTAB.MoveFirst
, Listbool = False

```

```

, Do Until RUXTAB.EOF
,   If (Printcounter = 0 And RUXTAB.Fields(6) = "No") Or (Printcounter = 1 And RUXTAB.Fields(6) = "Yes") Then
,     If RUXTAB.Fields(0) = CGI_AuthUser And RUXTAB.Fields(2) = Mptable And RUXTAB.Fields(3) = RUQuery.Fields(3) And
RUXTAB.Fields(4) = RUQuery.Fields(4) Then
,       If Listbool = False Then Send ("<ul>")
,       Listbool = True
,       Send ("<form method="post" action=""/win cgi/updt ru.exe">")
,       Send ("<INPUT TYPE="HIDDEN" NAME="SESSION" VALUE="" & Session & "">")
,       Send ("<INPUT TYPE="HIDDEN" NAME="MTPNAME" VALUE="" & MTPNAME & "">")
,       Send ("<INPUT TYPE="HIDDEN" NAME="FILE" VALUE="" & Filename & "">")
,       Send ("<INPUT TYPE="HIDDEN" NAME="MPTTABLE" VALUE="" & Mptable & "">")
,       Send ("<INPUT TYPE="HIDDEN" NAME="PTTASK" VALUE="" & Temptask & "">")
,       Send ("<INPUT TYPE="HIDDEN" NAME="PTTASKNUM" VALUE="" & RUQuery.Fields(3) & "">")
,       Send ("<INPUT TYPE="HIDDEN" NAME="RTABLE" VALUE="" & RUQuery.Fields(4) & "">")
,       Send ("<INPUT TYPE="HIDDEN" NAME="RUNAME" VALUE="" & RUQuery.Fields(6) & "">")
,       Send ("<INPUT TYPE="HIDDEN" NAME="RUTASK" VALUE="" & RUXTAB.Fields(9) & "">")
,       Send ("<INPUT TYPE="HIDDEN" NAME="RUTASKNUM" VALUE="" & RUXTAB.Fields(5) & "">")
,       Send ("<li><hr><font size=1><INPUT TYPE="SUBMIT" VALUE="RU Task - " & RUXTAB.Fields(9) & ""></font>")
,       If Printcounter = 1 And (RUXTAB.Fields(8) <> " " And RUXTAB.Fields(8) <> "") Then Send ("<br>Task Level Comments:" &
RUXTAB.Fields(8))
,       Send ("</form>")
,       End If
,     End If
,     If Printcounter = 0 And RUQuery.Fields(5) <> "Yes" And RUXTAB.Fields(6) = "Yes" And RUXTAB.Fields(0) = CGI_AuthUser And
RUXTAB.Fields(2) = Mptable And RUXTAB.Fields(3) = RUQuery.Fields(3) And RUXTAB.Fields(4) = RUQuery.Fields(4) Then
,       If Listbool = False Then
,         Send ("<ul>")
,         Listbool = True
,         End If
,       Send ("<li><hr><b> Completed Represented Unit Tasks - " & RUXTAB.Fields(9) & "</b><p>")
,       End If
,       RUXTAB.MoveNext
,       Loop
,     End If 'ruxtabbool
,     If Listbool = True And Ruxtabbool = False Then Send ("</ul>")
,     RUQuery.MoveNext
,     Loop
,   End If
,   Send ("</ul>")
,   Printcounter = Printcounter + 1
,   Loop
,   Send ("</HTML>")
,   Send ("</BODY>")
,   RUXTAB.Close
,   RUQuery.Close
,   RUdb.Close
,   xx = Freefile
,   Open "c:\netscape\server\docs\input\temp\testru3.txt" For Output As #xx
,   Print #xx, CGI_AuthUser & " "; Now & " " & Nowthen

```

Close #xx

End Sub

## APPENDIX G - SOURCE CODE FOR EVALUATION OF REPRESENTED UNIT TASKS (MAPRU.EXE)

```

Sub CGI_Main()
    Dim Gotit, Rubool As Boolean
    Dim i, x, y, l, m, px, Session, MtpTasknum, Pos As Integer
    Dim RUTASK, RUTASKNUM, PTTASKNUM, Filename, RUText, mptext, Listtype, MTPNAME, temptask, Runame, Lastelement, Mpttable, RUTable,
    PTTASKTEXT As String
    Dim Db, Db1, Userdb, RUdb As Database
    Dim Table, Table1, Usertable, RUxs As Recordset

    Send ("Content-type: text/html")
    Send ("")

    Set Db = OpenDatabase("c:\netscape\server\docs\input\database\mtptask.mdb")

    Set RUdb = OpenDatabase("c:\netscape\server\docs\input\database\mtpru.mdb")

    '
    ' Get the user's session number
    '
    i = 0

    Mpttable = "x"

    Do While i < CGI_NumFormTuples
        If LCase(CGI_FormTuples(i).key) = "session" Then Session = (CGI_FormTuples(i).value)
        If CGI_FormTuples(i).key = "FILE" Then Filename = (CGI_FormTuples(i).value)
        If CGI_FormTuples(i).key = "mpttasknum" Then MtpTasknum = Val(CGI_FormTuples(i).value)
        If CGI_FormTuples(i).key = "mtpselect" Then Mpttable = Left(CGI_FormTuples(i).value, 8)
        If CGI_FormTuples(i).key = "ruselect" Then RUTable = CGI_FormTuples(i).value
        If LCase(CGI_FormTuples(i).key) = "mtpname" Then MTPNAME = CGI_FormTuples(i).value
        If CGI_FormTuples(i).key = "SESSION" Then Session = Val(CGI_FormTuples(i).value)
        If LCase(CGI_FormTuples(i).key) = "ptatask" Then PTTASKTEXT = CGI_FormTuples(i).value
        If CGI_FormTuples(i).key = "mpttasknum" Then PTTASKNUM = CGI_FormTuples(i).value
        If CGI_FormTuples(i).key = "RUNAME" Then Runame = CGI_FormTuples(i).value
        If CGI_FormTuples(i).key = "RUTASKNUM" Then RUTASKNUM = CGI_FormTuples(i).value
        If CGI_FormTuples(i).key = "RUTASK" Then RUTASK = CGI_FormTuples(i).value
        i = i + 1
    Loop

    mptext = Mpttable

    ' Set RUxs = RUdb.OpenRecordset("RU_Xs")

    Rubool = False
    Set RUxs = RUdb.OpenRecordset("SELECT DISTINCTROW RU_Xs.USERID, RU_Xs.Session, RU_Xs.MTP, RU_Xs.MTP_Task, RU_Xs.RU, RU_Xs.RU_Task,
    RU_Xs.Update, RU_Xs.TSSC, RU_Xs.Comments, RU_Xs.RU_Task_Desc From RU_Xs WHERE ((RU_Xs.USERID Like " & Chr(39) & CGI_AuthUser & Chr(39)

```

```

& ") AND (RU_Xs.MTP Like " & Chr(39) & Mptable & Chr(39) & ") AND (RU_Xs.RU Like " & Chr(39) & RUTable & Chr(39) & ") ORDER BY
RU_Xs.MTP_Task, RU_Xs.RU_Task, RU_Xs.Update;"
If RUxs.EOF = True And RUxs.EOF = True Then Rubool = True
RUxs.Lockedits = False

Set Table = Db.OpenRecordset(RUTable, dbOpenSnapshot)

Send ("<HTML><CENTER><TITLE>Training Audience Task - " & PTASKTEXT & "</TITLE></CENTER>")
Send ("<BODY TEXT=""#000000"" BGCOLOR=""#00FFFF"" LINK=""#0000EE"" VLINK=""#551A8B"" ALINK=""#FF0000"">")
Send ("<br><H2><Center>Select Represented Unit Tasks for " & Runame & "<br>")
Send ("<FORM METHOD=""POST"" ACTION= ""/wincgi/ruxs.exe"" >")

Send ("Supporting Training Audience - " & MTPNAME & "<br>")
Send ("Conducting Training Audience Task - " & PTASKTEXT & "</center></h2>")

Send ("<Center><b>Select all of the Represented Unit tasks to support the Training Audience and its task in WARSIM:</B></center>")
Send ("<Center>When complete, push the button at the bottom of this frame.</center><p>")

Send ("<FORM METHOD=""POST"" ACTION= ""/wincgi/updtru.exe"" >")
Send ("<INPUT TYPE=""HIDDEN"" NAME=""FILE"" VALUE="" " & Filename & ""> ")
Send ("<INPUT TYPE=""HIDDEN"" NAME=""SESSION"" VALUE="" " & Session & ""> ")
Send ("<INPUT TYPE=""HIDDEN"" NAME=""mptable"" VALUE="" " & Mptable & ""> ")
Send ("<INPUT TYPE=""HIDDEN"" NAME=""mtpname"" VALUE="" " & MTPNAME & ""> ")
Send ("<INPUT TYPE=""HIDDEN"" NAME=""rutable"" VALUE="" " & RUTable & ""> ")
Send ("<INPUT TYPE=""HIDDEN"" NAME=""PTASKNUM"" VALUE="" " & MTPTasknum & ""> ")

Send ("<TABLE BORDER=1>")
Send ("<TR><TD><B><Center>Task Number</H3></B></TD><B><Center>Task <br> Conditions<br>
Standards</center></H3></B></TD><B><Center>Represented Unit TASK?</center></H3></B></TD></TR>")
Do Until Table.EOF

If Table.Fields(1) <> "" And Table.Fields(1) <> Lastelement Then Send ("<tr><td></td><td><font size=+1><B>ELEMENT: </B> " +
Table.Fields(1) + "</font></td><td></td></tr>")
Lastelement = Table.Fields(1)

If Table.Fields(2) <> "" Then
Send ("<TR><TD><font size=+1> " & Table.Fields(2) + "</font></TD><TD>")
Else
Send ("<TR><TD></TD><TD>")
End If

If Table.Fields(3) <> "" Then
temptask = Table.Fields(3)
Pos = Instr(1, temptask, "(" , 1)
If Pos = 0 Then Pos = Len(Table.Fields(3)) + 1
temptask = Left$(temptask, Pos - 1)
Send ("<B><font size=+1>TASK: </font></B> " & temptask)
End If

```



```

' Send ("<INPUT TYPE=""HIDDEN"" NAME=""PTATask"" VALUE="" " & PTATASKTEXT & "">")
' Send ("<INPUT TYPE=""HIDDEN"" NAME=""PTATASKNUM"" VALUE="" " & MTPTASKNUM & "">")
' Send ("<INPUT TYPE=""HIDDEN"" NAME=""RUTABLE"" VALUE="" " & RUTABLE & "">")
' Send ("<INPUT TYPE=""HIDDEN"" NAME=""RUNAME"" VALUE="" " & Runame & "">")
' Send ("<INPUT TYPE=""HIDDEN"" NAME=""RUTASK"" VALUE="" " & temptask & "">")
' Send ("<INPUT TYPE=""HIDDEN"" NAME=""RUTASKNUM"" VALUE="" " & Table.Fields(6) & "">")

' Send ("<td><INPUT TYPE=""HIDDEN"" VALUE= "RU Task - " & temptask & ""></td></TR>")
' Send ("<td><INPUT TYPE=""SUBMIT"" VALUE= "RU Task - " & temptask & ""></td></form></TR>")
' Send ("<td></td>")
' Send ("</TD></TR>")
' Send ("</tr>")

End If

Table.MoveNext
Loop
Send ("</table>")
Send ("<><Center><INPUT TYPE=SUBMIT VALUE= "Return to RU Task Selection/ Evaluation" "><INPUT TYPE= "RESET" VALUE= "Clear
Current Responses"></Center>")
Send ("</form>")
Table.Close
RUXs.Close
RUDb.Close

Send ("</HTML>")
Send ("</BODY>")
Db.Close

```

## APPENDIX H - SOURCE CODE FOR DETAILED SELECTION OF REPRESENTED UNIT TASKS

(MAPRU2.EXE)

```

Sub CGI_Main()
    Dim Gotit, Rubool As Boolean
    Dim i, X, Y, L, m, px, Session, MTPTasknum, Pos As Integer
    Dim RUTASK, RUTASKNUM, PTATASKNUM, Filename, Rutext, mptext, Listtype, MTPNAME, temptask, Runame, Lastelement, Mtptable, RUTable,
    PTATASKTEXT As String
    Dim Db, Db1, Userdb, RUdb As Database
    Dim Table, Table1, Usertable, RUxs As Recordset

    Send ("Content-type: text/html")
    Send ("")

    Set Db = OpenDatabase("c:\netscape\server\docs\input\database\mtptask.mdb")
    Set RUdb = OpenDatabase("c:\netscape\server\docs\input\database\mtpru.mdb")

    '
    ' Get the user's session number
    '
    i = 0
    Mtptable = "x"

    Do While i < CGI_NumFormTuples
        If LCase(CGI_FormTuples(i).key) = "session" Then Session = CGI_FormTuples(i).value
        If CGI_FormTuples(i).key = "FILE" Then Filename = CGI_FormTuples(i).value
        If CGI_FormTuples(i).key = "mtptasknum" Then MTPtasknum = Val(CGI_FormTuples(i).value)
        If CGI_FormTuples(i).key = "mtpselect" Then Mtptable = Left(CGI_FormTuples(i).value, 8)
        If CGI_FormTuples(i).key = "ruselect" Then RUTable = CGI_FormTuples(i).value
        If LCase(CGI_FormTuples(i).key) = "mtpname" Then MTPNAME = CGI_FormTuples(i).value
        If CGI_FormTuples(i).key = "SESSION" Then Session = Val(CGI_FormTuples(i).value)
        If LCase(CGI_FormTuples(i).key) = "ptatask" Then PTATASKTEXT = CGI_FormTuples(i).value
        If CGI_FormTuples(i).key = "mtptasknum" Then PTATASKNUM = CGI_FormTuples(i).value
        If CGI_FormTuples(i).key = "RUNAME" Then Runame = CGI_FormTuples(i).value
        If CGI_FormTuples(i).key = "RUTASKNUM" Then RUTASKNUM = CGI_FormTuples(i).value
        If CGI_FormTuples(i).key = "RUTASK" Then RUTASK = CGI_FormTuples(i).value
        i = i + 1
    Loop
    mptext = Mtptable

    ' Set RUxs = RUdb.OpenRecordset ("RU_Xs")

    Rubool = False

```

```

Set RUxs = RUdb.OpenRecordset("SELECT DISTINCTROW RU_Xs.USERID, RU_Xs.Session, RU_Xs.MTP_Task, RU_Xs.RU, RU_Xs.RU_Task,
RU_Xs.Update, RU_Xs.TSFC, RU_Xs.Comments, RU_Xs.RU_Task_Desc From RU_Xs WHERE ((RU_Xs.USERID Like " & Chr(39) & CGI_AuthUser & Chr(39)
& ") AND (RU_Xs.MTP_Like " & Chr(39) & MtpTable & Chr(39) & ") AND (RU_Xs.RU Like " & Chr(39) & RUTable & Chr(39) & " ))ORDER BY
RU_Xs.MTP_Task, RU_Xs.RU_Task, RU_Xs.Update;")
If RUxs.EOF = True And RUxs.EOF = True Then Rubool = True
RUxs.Lockedits = False

Set Table = Db.OpenRecordset(RUTable, dbOpenSnapshot)

Send ("<HTML><CENTER><TITLE>Training Audience Task - " & PTATASKTEXT & "</TITLE></CENTER>")
Send ("<BODY TEXT="#000000" BGCOLOR="#00FFFF" LINK="#0000EE" VLINK="#551A8B" ALINK="#FF0000">")
Send ("<br><H2><Center>Evaluate Represented Unit Tasks for " & Runame & "<br>")
Send ("Supporting Training Audience - " & MTPNAME & "<br>")
Send ("Conducting Training Audience Task - " & PTATASKTEXT & "</center></h2>")

Send ("<Center><br>Evaluate all of the selected Represented Unit tasks to support the Training Audience and its task in
WARSIM:</B></center>")

Send ("<TABLE BORDER=1>")
Send ("<TR><TD><B><H3><Center>Task Number</H3></B></TD><TD><B><H3><Center>Task <br> Conditions<br>
Standards</center></H3></B></TD><TD><B><H3><Center>Previously Evaluated?</center></H3></B></TD><TD><B><H3><Center>Evaluate RU
TASK</center></H3></B></TD></TR>")
Do Until Table.EOF

'
' Here is where I figure out is a guy X'd the block - will need to redo this with the appropriate table

Gotit = False
If Rubool = False Then
    RUxs.MoveFirst
    Do Until RUxs.EOF

        If RUxs.Fields(0) = CGI_AuthUser And RUxs.Fields(2) = MtpTable And RUxs.Fields(4) = RUTable And RUxs.Fields(3) = MTPTasknum
And RUxs.Fields(5) = Table.Fields(6) Then

            If Table.Fields(1) <> " " And Table.Fields(1) <> Lastelement Then Send ("<tr><td></td><td><font size=+1><B>ELEMENT: </B> " +
Table.Fields(1) + "</font></td><td></td></tr>")
            Lastelement = Table.Fields(1)

            If Table.Fields(2) <> " " Then
                Send ("<TR><TD><font size=+1> " & Table.Fields(2) + "</font></TD><TD>")
            Else

```

```

Send ("<TR><TD></TD><TD>")
End If

If Table.Fields(3) <> "" Then
    temptask = Table.Fields(3)
    Pos = Instr(1, temptask, "( ", 1)
    If Pos = 0 Then Pos = Len(Table.Fields(3)) + 1
    temptask = Left$(temptask, Pos - 1)
    Send ("<B><font size=+1>TASK: </font></B>" + temptask)
End If

If Table.Fields(6) <> "" Then
    Send ("<br></font></td><td><center>")
    If RUxs.Fields(6) = "Yes" Then
        Send ("Yes")
    Else
        Send ("No")
    End If
    Send ("</center></h3></td><FORM METHOD=""POST"" ACTION= ""'/wincgi/updttru.exe"" >")
End If

Send ("<INPUT TYPE=""HIDDEN"" NAME=""SESSION"" VALUE="" "" & Session & """>")
Send ("<INPUT TYPE=""HIDDEN"" NAME=""MTPNAME"" VALUE="" "" & MTPNAME & """>")
Send ("<INPUT TYPE=""HIDDEN"" NAME=""FILE"" VALUE="" "" & Filename & """>")
Send ("<INPUT TYPE=""HIDDEN"" NAME=""MTPTABLE"" VALUE="" "" & Mtptable & """>")
Send ("<INPUT TYPE=""HIDDEN"" NAME=""PTATASK"" VALUE="" "" & PTATASKTEXT & """>")
Send ("<INPUT TYPE=""HIDDEN"" NAME=""PTATAKNUM"" VALUE="" "" & MTPtasknum & """>")
Send ("<INPUT TYPE=""HIDDEN"" NAME=""RUTABLE"" VALUE="" "" & Rutable & """>")
Send ("<INPUT TYPE=""HIDDEN"" NAME=""RUNAME"" VALUE="" "" & Runame & """>")
Send ("<INPUT TYPE=""HIDDEN"" NAME=""RUTASK"" VALUE="" "" & temptask & """>")
Send ("<INPUT TYPE=""HIDDEN"" NAME=""RUTASKNUM"" VALUE="" "" & RUxs.Fields(5) & """>")

Send ("<td><INPUT TYPE=""SUBMIT"" VALUE= ""RU Task - " & temptask & """></td></form></TR>")
Gotit = True
RUxs.MoveLast
End If
RUxs.MoveNext
Loop
End If

If Gotit = False Then
    If Table.Fields(6) <> "" Then Send ("<br></font></td><td><FORM METHOD=""POST"" ACTION= ""'/wincgi/updttru.exe"" ><Center><INPUT type=""checkbox"></td><INPUT TYPE=""HIDDEN"" NAME=""ruselect"" & Rutable & "" value="" "" & MTPtasknum & "" & (Table.Fields(6) & "" ">")
    Send ("<INPUT TYPE=""HIDDEN"" NAME=""MTPNAME"" VALUE="" "" & MTPNAME & """>")
    Send ("<INPUT TYPE=""HIDDEN"" NAME=""FILE"" VALUE="" "" & Filename & """>")
    Send ("<INPUT TYPE=""HIDDEN"" NAME=""MTPTABLE"" VALUE="" "" & Mtptable & """>")
    Send ("<INPUT TYPE=""HIDDEN"" NAME=""PTATASK"" VALUE="" "" & PTATASKTEXT & """>")
    Send ("<INPUT TYPE=""HIDDEN"" NAME=""PTATAKNUM"" VALUE="" "" & MTPtasknum & """>")
    Send ("<INPUT TYPE=""HIDDEN"" NAME=""RUTABLE"" VALUE="" "" & Rutable & """>")
    Send ("<INPUT TYPE=""HIDDEN"" NAME=""RUNAME"" VALUE="" "" & Runame & """>")

```

```

'      Send ("<INPUT TYPE="HIDDEN" NAME="RUTASK" VALUE="" & temptask & "">")
'      Send ("<INPUT TYPE="HIDDEN" NAME="RUTASKNUM" VALUE="" & Table.Fields(6) & "">")
'      Send ("<td><INPUT TYPE="SUBMIT" VALUE="RU Task - " & temptask & ""></td></form></TR>")

      End If

      Table.MoveNext
      Loop
      Send ("</table>")
      Send ("<P><Center><INPUT TYPE=SUBMIT VALUE="Return to RU Task Selection/ Evaluation" ><INPUT TYPE="RESET" VALUE="Clear
      Current Responses"></Center>")
      Send ("</form>")
      Table.Close
      RUXs.Close
      RUdb.Close

      Send ("</HTML>")
      Send ("</BODY>")
      Db.Close

```

# APPENDIX I - SOURCE CODE FOR PRESENTATION OF REPRESENTED UNIT TASK DETAILS

(UPDTRU.EXE)

```

Sub CGI_Main()
Dim Filenumber, Pos, Ptatasknum, Rutasknum, i As Integer
Dim RUBool, TSSCDatabool, TSCB00L2 As Boolean
Dim Db, Db1, TSSCdb, RUBd As Database
Dim Table, Table1, TSSCData, RUdata As Recordset
Dim Rating, tempfile, Teststring, Comments, Prodpipe, Prodpipe2, Prodpipe3, temptask, PTATASK, RUTable, RUTask, RUName, Filename As String
Dim Nextrec, xx As Integer
Dim Currentfield2, Currentfield4, Currentfield5, Currentfield6 As Integer
i = 0

Do While i < CGI_NumFormTuples
    If CGI_FormTuples(i).key = "SESSION" Then Session = CGI_FormTuples(i).value
    If CGI_FormTuples(i).key = "MTPNAME" Then Mtpname = CGI_FormTuples(i).value
    If CGI_FormTuples(i).key = "MPTABLE" Then Mptable = CGI_FormTuples(i).value
    If LCase(CGI_FormTuples(i).key) = "ptatask" Then PTATASK = CGI_FormTuples(i).value
    If CGI_FormTuples(i).key = "PTATASKNUM" Then Ptatasknum = CGI_FormTuples(i).value
    If CGI_FormTuples(i).key = "RUTABLE" Then RUTable = CGI_FormTuples(i).value
    If CGI_FormTuples(i).key = "RUNAME" Then RUName = CGI_FormTuples(i).value
    If CGI_FormTuples(i).key = "RUTASK" Then Rutask = CGI_FormTuples(i).value
    If CGI_FormTuples(i).key = "RUTASKNUM" Then Rutasknum = CGI_FormTuples(i).value
    If CGI_FormTuples(i).key = "FILE" Then Filename = CGI_FormTuples(i).value
    i = i + 1
Loop

Set Db = OpenDatabase("c:\netscape\server\docs\input\database\mtptask.mdb")
Set Db1 = OpenDatabase("c:\netscape\server\docs\input\database\mtppm.mdb")
Set Table = Db.OpenRecordset(RUTable, dbOpenSnapshot)
Set Table1 = Db1.OpenRecordset(RUTable, dbOpenSnapshot)
Set TSSCdb = OpenDatabase("c:\netscape\server\docs\input\database\mtpru.mdb")

TSSCDatabool = False
Set TSSCData = TSSCdb.OpenRecordset("SELECT DISTINCTROW " & RUTable & ".USERID, " & RUTable & ".Session, " & RUTable & ".MTP, " & RUTable & ".MTP_Task, " & RUTable & ".RU_Task, " & RUTable & ".RU_PM, " & RUTable & ".Rating, " & RUTable & ".Equip, " & RUTable & ".Object, " & RUTable & ".Comment From " & RUTable & " Where ((" & RUTable & ".USERID Like "" & CGI_AuthUser & """) And (" & RUTable & ".MTP Like "" & Mptable & """) And (" & RUTable & ".MTP_Task = " & Ptatasknum & ") and (" & RUTable & ".RU_Task = " & Rutasknum & ")")
, dbOpenSnapshot)
If TSSCData.EOF = True And TSSCData.BOF = True Then TSSCDatabool = True
Set TSSCData = TSSCdb.OpenRecordset(RUTable, dbOpenSnapshot)
Table.FindFirst "[Counter] =" & Rutasknum

temptask = Table.Fields(3)
Pos = Instr(1, temptask, "(" , 1)
If Pos = 0 Then Pos = Len(Table.Fields(3)) + 1
temptask = Left$(temptask, Pos - 1)

```

```

Send ("Content-type: text/html")
Send ("")
Send ("<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML//EN">")
Send ("<HTML><HEAD><TITLE>TA Task - " & PTATASK & " </TITLE>")

Send ("<SCRIPT LANGUAGE="JavaScript">") '<!-- Beginning of JavaScript Applet -----'
Send ("<!-- Hide")

Send ("var scrTxt="Evaluate " & RUTask & " conducted by " & RUName & " as a Represented Unit for Training Audience " & Mtpname & "
conducting task " & PTATASK & ". ";")
Send ("var lenTxt=scrTxt.length;")
Send ("var width=200;")
Send ("var pos=1-width;")

Send ("function scroll() {")
Send ("pos++;")
Send ("var scroller="";")
Send ("if (pos==lenTxt) {")
Send ("pos=1-width;")
Send ("}")
Send ("if (pos<0) {")
Send ("for (var i=1; i<=Math.abs(pos); i++) {")
Send ("scroller=scroller+"";")
Send ("scroller=scroller+scrTxt.substring(0,width-i+1);")
Send ("}")
Send ("else {")
Send ("scroller=scroller+scrTxt.substring(pos,width+pos);")
Send ("}")
Send ("window.status = scroller;")
Send ("setTimeout("scroll()",150);")
Send ("}")
Send ("//-->")
Send ("</SCRIPT>")

Send ("<BODY bgcolor="00FFFF" text="000000" onLoad="scroll();return true;">")

Send ("<H2><Center> Rate Steps and Performance Measures </Center></H2>")
Send ("<H2><Center> Represented Unit - " & RUName & " <br>Performing Task " & RUTask & " <br>in support of " & Mtpname & "
<br>conducting task " & PTATASK & "</Center></H2>")

Send ("<FORM METHOD= "POST" ACTION="/wincci/ruxs.exe">")
Send ("<INPUT TYPE="HIDDEN" NAME="SESSION" VALUE="" & Session & "">")
Send ("<INPUT TYPE="HIDDEN" NAME="MTPNAME" VALUE="" & Mtpname & "">")
Send ("<INPUT TYPE="HIDDEN" NAME="MPTTABLE" VALUE="" & MPTTable & "">")
Send ("<INPUT TYPE="HIDDEN" NAME="PTATASK" VALUE="" & PTATASK & "">")

```

```

Send ("<INPUT TYPE=""HIDDEN"" NAME=""PTASKNUM"" VALUE="" & Ptasknum & "">")
Send ("<INPUT TYPE=""HIDDEN"" NAME=""RUTABLE"" VALUE="" & RUTABLE & "">")
Send ("<INPUT TYPE=""HIDDEN"" NAME=""RUNAME"" VALUE="" & RUNAME & "">")
Send ("<INPUT TYPE=""HIDDEN"" NAME=""RUTASK"" VALUE="" & RUTASK & "">")
Send ("<INPUT TYPE=""HIDDEN"" NAME=""RUTASKNUM"" VALUE="" & Rutasknum & "">")
Send ("<INPUT TYPE=""HIDDEN"" NAME=""FILE"" VALUE="" & Filename & "">")
Send ("<INPUT TYPE=""HIDDEN"" NAME=""RUTASKSBOOL"" VALUE="" & Filename & "">")
Send ("<IMAGE SRC=""/input/images/rain_lin.gif"" width=100%>")
Send ("<table border=1>")
If Table.Fields(2) <> "" Then Send ("<td><font size=+1><B>TASK NUMBER: </B></font>" + Table.Fields(2) + "<br>")
If Table.Fields(1) <> "" Then Send ("<font size=+1><B>ELEMENT: </B></font>" + Table.Fields(1) + "<br>")
If Table.Fields(3) <> "" Then Send ("<font size=+1><B>TASK: </B></font>" + Table.Fields(3) + "<br>")
If Table.Fields(4) <> "" Then Send ("<font size=+1><B>CONDITIONS: </B></font>" + Table.Fields(4) + "<br>")
If Table.Fields(5) <> "" Then Send ("<font size=+1><B>STANDARDS: </B></font>" + Table.Fields(5) + "<br>")
Send ("<P>")
Send ("</td><td align=left valign=bottom><b>TSSC TASK STEP/ PERFORMANCE MEASURE RATINGS</b><p><p> Given this Represented Unit performs
this task in support of your Training Audience task, to what extent do you require the details of the accomplishment of this step/
performance measure?<p>")
Send ("<U><b>FULL DETAILS:</b></U><U>All aspects of the RU's TS/ PM must be modeled in the simulation to stimulate the associated
Training Audience Task.<br>")
Send ("<U><b>PARTIAL DETAILS:</b></U><sup>*</sup><U>Only essential aspects of this RU's TS/ PM must be modeled in the simulation to
stimulate the associated Training Audience Task.<br>")
Send ("<U><b>MINIMUM DETAILS:</b></U><sup>*</sup><U>All this is required is to know the the RU TS/ PM has been executed (Yes/No).<br>")
Send ("<U><b>NOT REQUIRED:</b></U><sup>*</sup><U>No Information Required.<p>")
Send ("<sup>*</sup><i>Indicates that comments are required.</i></B></td><td align=center valign=bottom><b> Enter Represented Unit
Major End Items of Equipment and Simulation Synthetic Environment Objects. <p> Use the following format:<br><i>item,
item</i><p>Example: bridge,river </B></td></tr>")
TaskNum = Table.Fields(6)
Table.Close
Table.FindFirst "[MPTaskReference] =" & TaskNum
Do Until Table.EOF
Table.MoveNext
If Not Table.EOF Then
If Table.Fields(4) = TaskNum Then
Nextrec = Table.Fields(2)
Else
Nextrec = 0
End If
Else
Nextrec = 0
End If
Table.MovePrevious
Currentfield2 = Table.Fields(2)
Currentfield4 = Table.Fields(4)
Currentfield5 = Table.Fields(5)
Currentfield6 = Table.Fields(6)
If Table.Fields(4) = TaskNum Then
If Currentfield2 = 4 Then Send ("<TD><B>" + Table.Fields(3) + "</B>")
If Currentfield2 = 5 Then Send ("<TD><B>" + Chr(187) + Table.Fields(3) + "</B>")
If Currentfield2 = 6 Then Send ("<TD><B>" + Chr(187) + Table.Fields(3) + "</B>")

```

```

If (Currentfield2 = 4 And Nextrec = 5) Or Currentfield2 = 6 Then
    i = i
Else
    Rating = "default"
    Prodpipe = " "
    Prodpipe2 = " "
    Prodpipe3 = " "

If TSSCDatabool = False Then
    TSSCData.MoveFirst

    '      xx = FreeFile
    '      Open "c:\netscape\server\docs\input\temp\test14.txt" For Output As #xx

    Do Until TSSCData.EOF

        If TSSCData.Fields(0) = CGI_AuthUser And TSSCData.Fields(2) = MPTTable And Val(TSSCData.Fields(3)) = Ptatasknum And
        Currentfield4 = Val(TSSCData.Fields(4)) And Currentfield5 = Val(TSSCData.Fields(5)) And Currentfield6 = Val(TSSCData.Fields(6)) Then

            '      Print #xx, TSSCData.Fields(2) & " " & TSSCData.Fields(3) & " " & TSSCData.Fields(4) & " " & TSSCData.Fields(7)

            Rating = TSSCData.Fields(7)
            Print #xx, Rating
            If TSSCData.Fields(8) <> "" Then
                Prodpipe = TSSCData.Fields(8)
            Else
                Prodpipe = " "
            End If
            Print #xx, Prodpipe & " 1"
            If TSSCData.Fields(9) <> "" Then
                Prodpipe2 = TSSCData.Fields(9)
            Else
                Prodpipe2 = " "
            End If
            Print #xx, Prodpipe2 & " 2"
            If TSSCData.Fields(10) <> "" Then
                Prodpipe3 = TSSCData.Fields(10)
            Else
                Prodpipe3 = " "
            End If
            Print #xx, Prodpipe3 & " 3"

```

```

End If
TSSCData.MoveNext
Loop
End If
' will need to do same for this field
' Close #xx

' xx = FreeFile
' Open "c:\netscape\server\docs\input\temp\test15.txt" For Output As #xx
' Print #xx, Mtpname
' Close #xx

Send ("</td><td width=30%>")
Send ("<INPUT type = "radio" name = "tssc-" & Currentfield4 & "-" & Currentfield5 & "-" & Currentfield6 & " "value="high" ")
If Rating = "high" Then
Send ("checked>Full Details<br>")
Else
Send (">Full Details<br>")
End If
Send ("<INPUT type = "radio" name = "tssc-" & Currentfield4 & "-" & Currentfield5 & "-" & Currentfield6 & " "value="med" ")
If Rating = "med" Then
Send ("checked>Partial Details<sup>*</sup><br>")
Else
Send (">Partial Details<sup>*</sup><br>")
End If
Send ("<INPUT type = "radio" name = "tssc-" & Currentfield4 & "-" & Currentfield5 & "-" & Currentfield6 & " "value="low" ")
If Rating = "low" Then
Send ("checked>Minimum Details<sup>*</sup><br>")
Else
Send (">Minimum Details<sup>*</sup><br>")
End If
Send ("<INPUT type = "radio" name = "tssc-" & Currentfield4 & "-" & Currentfield5 & "-" & Currentfield6 & " "value="none" ")
If Rating = "none" Or Rating = "default" Then
Send ("checked>Not Required<sup>*</sup><br>")
Else
Send (">Not Required<sup>*</sup><br>")
End If
Send ("<sup>*</sup><sup>Comments:<br><input type=Text name = "cmts-" & Currentfield4 & "-" & Currentfield5 & "-" & Currentfield6 & " "value=" " & Prodpipe3 & " " Size = 20 ></td>")
Send ("<td align=center>Equipment:<br><input type=Text name = "equip-" & Currentfield4 & "-" & Currentfield5 & "-" & Currentfield6 & " "value=" " & Prodpipe2 & " " Size = 20 ><br>")
Send ("SE Objects:<br><input type=Text name = "objs-" & Currentfield4 & "-" & Currentfield5 & "-" & Currentfield6 & " "value=" " & Prodpipe2 & " " Size = 20 ></td>")

End If
Send ("</td></tr>")
End If

If Table1.Fields(4) <> Tasknum Then GoTo bob

```

```

Table1.MoveNext
Loop
bob:
Send ("</table>")
,
Table1.Close
TSSCData.Close

TSSCBOOL2 = False
Set TSSCData = TSSCdb.OpenRecordset("SELECT DISTINCTROW RU_Xs.USERID, RU_Xs.Session, RU_Xs.MTP, RU_Xs.MTP_Task, RU_Xs.RU,
RU_Xs.RU_Task, RU_Xs.Update, RU_Xs.TSSC, RU_Xs.Comments, RU_Xs.RU_Task_Desc From RU_Xs WHERE ((RU_Xs.USERID Like " & Chr(39) &
CGI_AuthUser & Chr(39) & ") AND (RU_Xs.MTP Like " & Chr(39) & MTPTable & Chr(39) & ") AND (RU_Xs.RU Like " & Chr(39) & RUTable &
Chr(39) & ") ORDER BY RU_Xs.MTP_Task, RU_Xs.RU_Task, RU_Xs.Update;")
If TSSCData.EOF = True And TSSCData.EOF = True Then TSSCBOOL2 = True
, TSSCData.LockEdits = False

, Set TSSCData = TSSCdb.OpenRecordset("RU_Xs")

Comments = " "

If TSSCBOOL2 = False Then

TSSCData.MoveFirst

Do Until TSSCData.EOF
, TSSCData.Fields(6) = "Yes"

If TSSCData.Fields(8) <> " " And TSSCData.Fields(0) = CGI_AuthUser And TSSCData.Fields(2) = MTPTable And Val(TSSCData.Fields(3)) =
Platasknum And TSSCData.Fields(4) = RUTable And Val(TSSCData.Fields(5)) = Rutasknum Then
Comments = TSSCData.Fields(8)

End If
TSSCData.MoveNext
Loop
End If

Send ("</SELECT></center>")
Send ("<P><center><font size =1>Provide any additional information that will assist the WARSIM 2000 simulation developer modeling
this Represented Unit in support of this particular Training Audience Task.</font></center><p>")
Send ("<center><TEXTAREA NAME=" "COMMENTS" Rows=5 Cols=50>" & Comments & "</Textarea></center><p>")
Send ("<P><Center><INPUT TYPE=" "SUBMIT" VALUE= "Submit Ratings" ><Input type="Reset" Value="Clear Current
Changes"></Center>")

Send ("</FORM>")
Send ("</BODY>")
Send ("</HTML>")

```

```
Db1.Close  
Db.Close  
TSCdata.Close  
TSCdb.Close  
End Sub
```

## APPENDIX J - SOURCE CODE TO BUILD QUERY ENGINE (QUERY.EXE)

```
Sub CGI_Main()
```

```
Dim Db, TPSCdb, RUdb, Userdb As Database
Dim MTPTable, Userdata As Recordset
```

```
Set Db = OpenDatabase("c:\netscape\server\docs\input\database\mtptask.mdb")
Set MTPTable = Db.OpenRecordset("MTPDIR", dbReadOnly)
Set TPSCdb = OpenDatabase("c:\netscape\server\docs\input\database\mtptpsc.mdb")
Set RUdb = OpenDatabase("c:\netscape\server\docs\input\database\mtpru.mdb")
```

```
Set Userdb = OpenDatabase("c:\netscape\server\docs\database\users.mdb")
Set Userdata = Userdb.OpenRecordset("Data", dbReadOnly)
Send ("Content-type: text/html")
Send ("")
```

```
Send ("<BODY bgcolor="#00FFFF" text="#000000">")
Send ("<HTML><HEAD><TITLE> WARSIM 2000 Query Engine Input Form</TITLE>")
Send ("<H2><Center> WARSIM 2000 Query Engine Input Form </Center></H2></HEAD>")
Send ("<FORM METHOD= "POST" ACTION="/win.cgi/runquery.exe">")
```

```
Send ("<IMAGE SRC="/input/images/rain_lin.gif" width=100%>")
```

```
Send ("<center>This query form will present data as it currently resides in the WARSIM 2000 database. ")
```

```
Send ("Select how you want to look at the data. You may look at all or any combination of users, Training Audiences and Represented Units.")
```

```
Send ("Feel free to experiment until you find what you are looking for. More queries are in the design stage now.")
```

```
Send ("I am also looking for any comments as to other queries that may be useful.")
```

```
Send ("<A HREF="mailto:fr0161@exmail.usma.edu">")
```

```
Send ("<ADDRESS>&lt;MAJ Bob Phelan&gt;</ADDRESS>")
```

```
Send ("</A>")
```

```
Send ("<IMAGE SRC="/input/images/rain_lin.gif" width=100%>")
```

```
, Build list of users
```

```
Send ("<center><B><h2> Select Which User(s) Data you want included in the Query</h2></b><p>")
```

```
Send ("<SELECT NAME = "USERS" MULTIPLE SIZE=10 >")
```

```
Send ("<OPTION SELECTED VALUE="ALL"><ALL>")
```

```
Userdata.MoveFirst
```

```
Do Until Userdata.EOF
```

```
    If Userdata.Fields(21) = True Then Send ("<OPTION VALUE="" & Userdata.Fields(6) & "">" & Userdata.Fields(1) & " " &
```

```
Userdata.Fields(0))
```

```
    Userdata.MoveNext
```

```
Loop
```

```
Send ("</SELECT>")
```

```
Send ("<center><B><h2> Select Which Training Audience(s) you want specifically included in the Query</h2></b><p>")
```

```
Send ("<SELECT NAME = "PTA" MULTIPLE SIZE=10 >")
```

```
Send ("<OPTION SELECTED VALUE="ALL"><ALL>")
```

```

MTPTTable.MoveFirst
Do Until MTPTTable.EOF
    Send ("<OPTION VALUE="" & Left(MTPTTable.Fields(1), 8) & "">" & MTPTTable.Fields(8))
    MTPTTable.MoveNext
Loop
Send ("</SELECT>")

Send ("<center><b><h2> Select Which Represented Unit(s) you want specifically included in the Query</h2></b><p>")
Send ("<SELECT NAME = "RU" MULTIPLE SIZE=10 >")
Send ("<OPTION SELECTED VALUE=""ALL"">ALL")
MTPTTable.MoveFirst
Do Until MTPTTable.EOF
    Send ("<OPTION VALUE="" & Left(MTPTTable.Fields(1), 8) & "">" & MTPTTable.Fields(8))
    MTPTTable.MoveNext
Loop
Send ("</SELECT><p>")

Send ("<Table border =2 width=100%><tr><th>Number</th><th><font size =+3>Submit A Query Type Below</font><br>Selected Query will
apply user options as appropriate.</th></tr>")
    Send ("<tr><th align = left><b><font size = +1>TA1:</th><th align=left>SME Work Status</th><th></tr>")
    Send ("<td><center><input type = submit name = "QUERY" Value="SME Work Status"></center></td></tr>")
    Send ("<input type=submit name="Set4" value="Run Specified Query">")

    Send ("<tr><th align = left><b><font size = +1>TA2:</th><th align=left>TPSCs by SME - Individual</th><th></tr>")
    Send ("<tr><th align = left><b><font size = +1>TA2:</th><th align=left>TPSCs by SME - Individual</th><th></tr>")
    Send ("<td><center><input type = "submit" name = "QUERY" Value="TPSCs by SME - Individual"></center></td></tr>")
    Send ("<td><center><input type = "radio" name = "QUERY" Value="TWO"></center></td></tr>")

    Send ("<tr><th align=left><b><font size = +1>TA3:</th><th align=left>TPSCs by SME - All</th><th></tr>")
    Send ("<tr><th align=left><b><font size = +1>TA3:</th><th align=left>TPSCs by SME - All</th><th></tr>")
    Send ("<td><center><input type = "submit" name = "QUERY" Value="TPSCs by SME - All"></center></td></tr>")
    Send ("<td><center><input type = "radio" name = "QUERY" Value="THREE"></center></td></tr>")

    Send ("<tr><th align=left><b><font size = +1>TA4:</th><th align=left>TPSCs by SME - All</th><th></tr>")
    Send ("<tr><th align=left><b><font size = +1>TA4:</th><th align=left>TPSCs by SME - All</th><th></tr>")
    Send ("<td><center><input type = "submit" name = "QUERY" Value="TPSCs by SME - All"></center></td></tr>")
    Send ("<td><center><input type = "radio" name = "QUERY" Value="FOUR"></center></td></tr>")

    Send ("<tr><th align=left><b><font size = +1>TA5:</th><th align=left>TPSCs by SME - All</th><th></tr>")
    Send ("<tr><th align=left><b><font size = +1>TA5:</th><th align=left>TPSCs by SME - All</th><th></tr>")
    Send ("<td><center><input type = "submit" name = "QUERY" Value="TPSCs by SME - All"></center></td></tr>")
    Send ("<td><center><input type = "radio" name = "QUERY" Value="FIVE"></center></td></tr>")

    Send ("<tr><th align = left><b><font size = +1>TA6:</th><th align=left>TPSCs by SME - All</th><th></tr>")
    Send ("<tr><th align = left><b><font size = +1>TA6:</th><th align=left>TPSCs by SME - All</th><th></tr>")
    Send ("<td><center><input type = "submit" name = "QUERY" Value="TPSCs by SME - All"></center></td></tr>")
    Send ("<td><center><input type = "radio" name = "QUERY" Value="SIX"></center></td></tr>")

    Send ("<tr><th align = left><b><font size = +1>TA7:</th><th align=left>TPSCs by SME - All</th><th></tr>")
    Send ("<tr><th align = left><b><font size = +1>TA7:</th><th align=left>TPSCs by SME - All</th><th></tr>")
    Send ("<td><center><input type = "submit" name = "QUERY" Value="TPSCs by SME - All"></center></td></tr>")
    Send ("<td><center><input type = "radio" name = "QUERY" Value="SEVEN"></center></td></tr>")

```

```

Send ("<td><center><input type = "submit" name = "QUERY" Value="TA Task Companion List">Long Query!</center></td></tr>")
Send ("<tr><th align = left><b>font size = +1>TA8:</th>")
Send ("<td><center><input type = "submit" name = "QUERY" Value="TA Task Companion List (Text Only)">Long
Query!</center></td></tr>")
Send ("<tr><th align = left><b>font size = +1>TA9:</th>")
Send ("<td><center><input type = "submit" name = "QUERY" Value="TA Task Exclusion List"></center></td></tr>")
Send ("<tr><td bgcolor="#FF0000"></td><td bgcolor="#FF0000"></td></tr>")
Send ("<tr><th align = left><b>font size = +1>RU1-1:</th>")
Send ("<tr><th align = left><b>font size = +1>RU1-1:</th><th align=left>RU frequency by TA Task</th>")
Send ("<td><center><input type = "submit" name = "QUERY" Value="RU frequency by TA Task"></center></td></tr>")
Send ("<td><center><input type = "radio" name = "QUERY" Value="FOURTEEN"></center></td></tr>")
Send ("<tr><th align = left><b>font size = +1>RU1-2:</th>")
Send ("<tr><th align = left><b>font size = +1>RU1-2:</th><th align=left>RUS by TA Task</th>")
Send ("<td><center><input type = "submit" name = "QUERY" Value="RUS by TA Task"></center></td></tr>")
Send ("<td><center><input type = "radio" name = "QUERY" Value="FOUR"></center></td></tr>")
Send ("<tr><th align = left><b>font size = +1>RU1-3:</th>")
Send ("<tr><th align = left><b>font size = +1>RU1-3:</th><th align=left>TA frequency by RU Task</th>")
Send ("<td><center><input type = "submit" name = "QUERY" Value="TA frequency by RU Task"></center></td></tr>")
Send ("<td><center><input type = "radio" name = "QUERY" Value="FIFTEEN"></center></td></tr>")
Send ("<tr><th align = left><b>font size = +1>RU1-4:</th>")
Send ("<tr><th align = left><b>font size = +1>RU1-4:</th><th align=left>TAS by RU Task</th>")
Send ("<td><center><input type = "submit" name = "QUERY" Value="TAS by RU Task"></center></td></tr>")
Send ("<td><center><input type = "radio" name = "QUERY" Value="FIVE"></center></td></tr>")
Send ("<tr><th align = left><b>font size = +1>RU2-1:</th>")
Send ("<tr><th align = left><b>font size = +1>RU2-1:</th><th align=left>TSSCs by SME - TA Task to RU Task</th>")
Send ("<td><center><input type = "submit" name = "QUERY" Value="TSSCs by SME - TA Task to RU Task"></center></td></tr>")
Send ("<tr><th align = left><b>font size = +1>RU2-2:</th>")
Send ("<tr><th align = left><b>font size = +1>RU2-2:</th><th align=left>TSSCs by SME - RU Task to TA Task</th>")
Send ("<td><center><input type = "submit" name = "QUERY" Value="TSSCs by SME - RU Task to TA Task"></center></td></tr>")
Send ("<td><center><input type = "radio" name = "QUERY" Value="EIGHT"></center></td></tr>")
Send ("<tr><th align = left><b>font size = +1>RU3:</th>")
Send ("<tr><th align = left><b>font size = +1>RU3:</th><th align=left>RU Task frequency by TA</th>")
Send ("<td><center><input type = "submit" name = "QUERY" Value="RU Task frequency by TA"></center></td></tr>")
Send ("<td><center><input type = "radio" name = "QUERY" Value="TEN"></center></td></tr>")
Send ("<tr><th align = left><b>font size = +1>RU4:</th>")
Send ("<tr><th align = left><b>font size = +1>RU4:</th><th align=left>TSSCs - All SMEs (Not ready - pending rollout
algorithm)</th>")
Send ("<td><center><input type = "submit" name = "QUERY" Value="TSSCs - All SMEs">(Do not use yet.)</center></td></tr>")
Send ("<td><center><input type = "radio" name = "QUERY" Value="THIRTEEN"></center></td></tr>")

```

```
Send ("<tr><th align = left><b><font size = +1>RU5-1:</th><b>"
```

```
' Send ("<tr><th align = left><b><font size = +1>RU5-1:</th><th align=left>TSSCs All SMEs - TA Task to RU Task</th><b>"
```

```
' Send ("<td><center><input type = "submit" name = "QUERY" Value=""TSSCs All SMEs - TA Task to RU Task"></center></td></tr><b>"
```

```
' Send ("<td><center><input type = "radio" name = "QUERY" Value=""ELEVEN"></center></td></tr><b>"
```

```
Send ("<tr><th align = left><b><font size = +1>RU5-2:</th><b>"
```

```
' Send ("<tr><th align = left><b><font size = +1>RU5-2:</th><th align=left>TSSCs All SMEs - RU Task to TA Task</th><b>"
```

```
' Send ("<td><center><input type = "submit" name = "QUERY" Value=""TSSCs All SMEs - RU Task to TA Task"></center></td></tr><b>"
```

```
' Send ("<td><center><input type = "radio" name = "QUERY" Value=""TWELVE"></center></td></tr><b>"
```

```
Send ("<tr><th align = left><b><font size = +1>RU6:</th><b>"
```

```
Send ("<td><center><input type = "submit" name = "QUERY" Value=""RU Exclusion List"></center></td></tr><b>"
```

```
Send ("<tr><th align = left><b><font size = +1>RU7:</th><b>"
```

```
Send ("<td><center><input type = "submit" name = "QUERY" Value=""RU Companion List"></center></td></tr><b>"
```

```
Send ("<tr><td BGColor=""#FF0000"></td><td BGColor=""#FF0000"></td><td BGColor=""#FF0000"></td></tr><b>"
```

```
Send ("<tr><th align = left><b><font size = +1>SE1:</th><b>"
```

```
Send ("<td><center><input type = "submit" name = "QUERY" Value=""Equipment - RU to RU Task"></center></td></tr><b>"
```

```
Send ("<tr><th align = left><b><font size = +1>SE2:</th><b>"
```

```
Send ("<td><center><input type = "submit" name = "QUERY" Value=""Equipment - TA Task to RU Task"></center></td></tr><b>"
```

```
yet!</center></td></tr><b>"
```

```
Send ("<tr><th align = left><b><font size = +1>SE3:</th><b>"
```

```
Send ("<td><center><input type = "submit" name = "QUERY" Value=""Environmental Objects - RU to RU Task"></center></td></tr><b>"
```

```
Send ("</tr><b>"
```

```
Send ("<tr><th align = left><b><font size = +1>SE4:</th><th align=left>Environmental Objects to RU Task to TA Task</th><b>"
```

```
Send ("</tr><b>"
```

```
Send ("</table><p>")
```

```
Send ("<input type=submit name="Set4" value="Run Specified Query">")
```

```

Send ("</FORM>")

If CGI_AuthUser = "mcnett" Then
    Send ("<FORM METHOD= "POST">" ACTION="/wincgi/recalc2.exe">")
    Send ("<td><center><input type = "submit" name = "RECALC" Value="RECALC"></center></td></tr>")
    Send "</form>"
    Send ("<FORM METHOD= "POST">" ACTION="/wincgi/recalcru.exe">")
    Send ("<td><center><input type = "submit" name = "RECALCRUS" Value="RECALCRUS"></center></td></tr>")
    Send "</form>"
End If
Send ("</BODY>")
Send ("</HTML>")

Userdata.Close
Userdb.Close
TPSCdb.Close
MTPTable.Close
Db.Close
Rldb.Close

End Sub

```

## APPENDIX K - SOURCE CODE FOR EXECUTION OF SELECTED QUERY (RUNQUERY.EXE)

```

Sub CGI_Main()

Dim Db, TPSCdb, RUdb, Userdb As Database
Dim MTPTable, Userdata As Recordset

Set Db = OpenDatabase("c:\netscape\server\docs\input\database\mtptask.mdb")
Set MTPTable = Db.OpenRecordset("MTPDIR", dbReadOnly)
Set TPSCdb = OpenDatabase("c:\netscape\server\docs\input\database\mtptpsc.mdb")
Set RUdb = OpenDatabase("c:\netscape\server\docs\input\database\mtpru.mdb")

Set Userdb = OpenDatabase("c:\netscape\server\docs\database\users.mdb")
Set Userdata = Userdb.OpenRecordset("Data", dbReadOnly)
Send ("Content-type: text/html")
Send ("")
Send ("<BODY bgcolor="#00FFFF" text="#000000">")
Send ("<HTML><HEAD><TITLE> WARSIM 2000 Query Engine Input Form</TITLE>")
Send ("<H2><Center> WARSIM 2000 Query Engine Input Form </Center></H2></HEAD>")
Send ("<FORM METHOD= "POST" ACTION="/wincgi/runquery.exe">")

Send ("<IMAGE SRC="/input/images/rain_lin.gif" width=100%>")
Send ("<center>This query form will present data as it currently resides in the WARSIM 2000 database. ")
Send ("Select how you want to look at the data. You may look at all or any combination of users, Training Audiences and Represented
Units. ")
Send ("Feel free to experiment until you find what you are looking for. More queries are in the design stage now.")
Send ("I am also looking for any comments as to other queries that may be useful.")
Send ("<A HREF="mailto:fr0161@exmail.usma.edu">")
Send ("<ADDRESS>&lt;MAJ Bob Phelan&gt;</ADDRESS>")
Send ("</A>")
Send ("<IMAGE SRC="/input/images/rain_lin.gif" width=100%>")
, Build list of users
,
Send ("<center><B><h2> Select Which User(s) Data you want included in the Query</h2></b><p>")
Send ("<SELECT NAME = "USERS" MULTIPLE SIZE=10 >")
Send ("<OPTION SELECTED VALUE="ALL">ALL")
Userdata.MoveFirst
Do Until Userdata.EOF
If Userdata.Fields(21) = True Then Send ("<OPTION VALUE=" & Userdata.Fields(6) & " "& Userdata.Fields(1) & " " &
Userdata.Fields(0))
Userdata.MoveNext
Loop
Send ("</SELECT>")

Send ("<center><B><h2> Select Which Training Audience(s) you want specifically included in the Query</h2></b><p>")
Send ("<SELECT NAME = "PTA" MULTIPLE SIZE=10 >")
Send ("<OPTION SELECTED VALUE="ALL">ALL")

```

```

MTPTTable.MoveFirst
Do Until MTPTTable.EOF
    Send ("<OPTION VALUE="" & Left(MTPTTable.Fields(1), 8) & "">" & MTPTTable.Fields(8))
    MTPTTable.MoveNext
Loop
Send ("</SELECT>")

Send ("<center><b><h2> Select Which Represented Unit(s) you want specifically included in the Query</h2></b><p>")
Send ("<SELECT NAME = "RU" MULTIPLE SIZE=10 >")
Send ("<OPTION SELECTED VALUE=""ALL"">ALL")
MTPTTable.MoveFirst
Do Until MTPTTable.EOF
    Send ("<OPTION VALUE="" & Left(MTPTTable.Fields(1), 8) & "">" & MTPTTable.Fields(8))
    MTPTTable.MoveNext
Loop
Send ("</SELECT><p>")

Send ("<Table border =2 width=100%><tr><th>Number</th><th><font size =+3>Submit A Query Type Below</font><br>Selected Query will
apply user options as appropriate.</th></tr>")
    Send ("<tr><th align = left><b><font size = +1>TA1:</th>")
    Send ("<tr><th align = left><b><font size = +1>TA1:</th><th align=left>SME Work Status</th>")
    Send ("<td><center><input type = submit name = "QUERY" Value="SME Work Status"></center></td></tr>")
    Send ("<input type=submit name="Set4" value="Run Specified Query">")

    Send ("<tr><th align = left><b><font size = +1>TA2:</th>")
    Send ("<tr><th align = left><b><font size = +1>TA2:</th><th align=left>TPSCs by SME - Individual</th>")
    Send ("<td><center><input type = "submit" name = "QUERY" Value="TPSCs by SME - Individual"></center></td></tr>")
    Send ("<td><center><input type = "radio" name = "QUERY" Value="TWO"></center></td></tr>")

    Send ("<tr><th align=left><b><font size = +1>TA3:</th>")
    Send ("<tr><th align=left><b><font size = +1>TA3:</th><th align=left>TPSCs by SME - All</th>")
    Send ("<td><center><input type = "submit" name = "QUERY" Value="TPSCs by SME - All"></center></td></tr>")
    Send ("<td><center><input type = "radio" name = "QUERY" Value="THREE"></center></td></tr>")

    Send ("<tr><th align=left><b><font size = +1>TA4:</th>")
    Send ("<tr><th align=left><b><font size = +1>TA4:</th><th align=left>TPSCs by SME - All</th>")
    Send ("<td><center><input type = "submit" name = "QUERY" Value="TPSCs by SME - All"></center></td></tr>")
    Send ("<td><center><input type = "radio" name = "QUERY" Value="TWO"></center></td></tr>")

    Send ("<tr><th align=left><b><font size = +1>TA5:</th>")
    Send ("<tr><th align=left><b><font size = +1>TA5:</th><th align=left>TPSCs by SME - All</th>")
    Send ("<td><center><input type = "submit" name = "QUERY" Value="TPSCs by SME - All"></center></td></tr>")
    Send ("<td><center><input type = "radio" name = "QUERY" Value="TWO"></center></td></tr>")

    Send ("<tr><th align=left><b><font size = +1>TA6:</th>")
    Send ("<td><center><input type = "submit" name = "QUERY" Value="TPSCs ordered by TPSC (v.2) - Individual w/
comments"></center></td></tr>")

    Send ("<tr><th align = left><b><font size = +1>TA7:</th>")

```

```

Send ("<td><center><input type = "submit"" name = "QUERY"" Value="TA Task Companion List"></center></td></tr>")
Send ("<tr><th align = left><b>font size = +1>TA8:</th>")
Send ("<td><center><input type = "submit"" name = "QUERY"" Value="TA Task Companion List (Text Only)"></tr>")
Query!</center></td></tr>")

Send ("<tr><th align = left><b>font size = +1>TA9:</th>")
Send ("<td><center><input type = "submit"" name = "QUERY"" Value="TA Task Exclusion List"></center></td></tr>")

Send ("<tr><td bgcolor="FF0000"></td><td bgcolor="FF0000"></td><td bgcolor="FF0000"></td></tr>")

Send ("<tr><th align = left><b>font size = +1>RU1-1:</th>")
Send ("<tr><th align = left><b>font size = +1>RU1-1:</th><th align=left>RU frequency by TA Task</th>")
Send ("<td><center><input type = "submit"" name = "QUERY"" Value="RU frequency by TA Task"></center></td></tr>")
Send ("<td><center><input type = "radio"" name = "QUERY"" Value="FOURTEEN"></center></td></tr>")

Send ("<tr><th align = left><b>font size = +1>RU1-2:</th>")
Send ("<tr><th align = left><b>font size = +1>RU1-2:</th><th align=left>RUS by TA Task</th>")
Send ("<td><center><input type = "submit"" name = "QUERY"" Value="RUS by TA Task"></center></td></tr>")
Send ("<td><center><input type = "radio"" name = "QUERY"" Value="FOUR"></center></td></tr>")

Send ("<tr><th align = left><b>font size = +1>RU1-3:</th>")
Send ("<tr><th align = left><b>font size = +1>RU1-3:</th><th align=left>TA frequency by RU Task</th>")
Send ("<td><center><input type = "submit"" name = "QUERY"" Value="TA frequency by RU Task"></center></td></tr>")
Send ("<td><center><input type = "radio"" name = "QUERY"" Value="FIFTEEN"></center></td></tr>")

Send ("<tr><th align = left><b>font size = +1>RU1-4:</th>")
Send ("<tr><th align = left><b>font size = +1>RU1-4:</th><th align=left>TAS by RU Task</th>")
Send ("<td><center><input type = "submit"" name = "QUERY"" Value="TAS by RU Task"></center></td></tr>")
Send ("<td><center><input type = "radio"" name = "QUERY"" Value="FIVE"></center></td></tr>")

Send ("<tr><th align = left><b>font size = +1>RU2-1:</th>")
Send ("<tr><th align = left><b>font size = +1>RU2-1:</th><th align=left>TSSCs by SME - TA Task to RU Task</th>")
Send ("<td><center><input type = "submit"" name = "QUERY"" Value="TSSCs by SME - TA Task to RU Task"></center></td></tr>")
Send ("<td><center><input type = "radio"" name = +1>RU2-2:</th>")
Send ("<tr><th align = left><b>font size = +1>RU2-2:</th><th align=left>TSSCs by SME - RU Task to TA Task</th>")
Send ("<td><center><input type = "submit"" name = "QUERY"" Value="TSSCs by SME - RU Task to TA Task"></center></td></tr>")
Send ("<td><center><input type = "radio"" name = "QUERY"" Value="EIGHT"></center></td></tr>")

Send ("<tr><th align = left><b>font size = +1>RU3:</th>")
Send ("<tr><th align = left><b>font size = +1>RU3:</th><th align=left>RU Task frequency by TA</th>")
Send ("<td><center><input type = "submit"" name = "QUERY"" Value="RU Task frequency by TA"></center></td></tr>")
Send ("<td><center><input type = "radio"" name = "QUERY"" Value="TEN"></center></td></tr>")

Send ("<tr><th align = left><b>font size = +1>RU4:</th>")
Send ("<tr><th align = left><b>font size = +1>RU4:</th><th align=left>TSSCs - All SMEs (Not ready - pending rollout algorithm)</th>")
Send ("<td><center><input type = "submit"" name = "QUERY"" Value="TSSCs - All SMEs"></center></td></tr>")
Send ("<td><center><input type = "radio"" name = "QUERY"" Value="THIRTEEN"></center></td></tr>")

```

```
' Send ("<tr><th align = left><b><font size = +1>RU5-1:</th><b>"
```

```
'   Send ("<tr><th align = left><b><font size = +1>RU5-1:</th><th align=left>tSSCs All SMES - TA Task to RU Task</th><b>"
```

```
'   Send ("<td><center><input type = "submit"> name = "QUERY" Value="tSSCs All SMES - TA Task to RU Task"></center></td></tr><b>"
```

```
'   Send ("<td><center><input type = "radio"> name = "EVEN" Value="ELEVEN"></center></td></tr><b>"
```

```
'   Send ("<tr><th align = left><b><font size = +1>RU5-2:</th><b>"
```

```
'   Send ("<tr><th align = left><b><font size = +1>RU5-2:</th><th align=left>tSSCs All SMES - RU Task to TA Task</th><b>"
```

```
'   Send ("<td><center><input type = "submit"> name = "QUERY" Value="tSSCs All SMES - RU Task to TA Task"></center></td></tr><b>"
```

```
'   Send ("<td><center><input type = "radio"> name = "TWELVE" Value="TWELVE"></center></td></tr><b>"
```

```
'   Send ("<tr><th align = left><b><font size = +1>RU6:</th><b>"
```

```
'   Send ("<td><center><input type = "submit"> name = "QUERY" Value="RU Exclusion List"></center></td></tr><b>"
```

```
'   Send ("<tr><th align = left><b><font size = +1>RU7:</th><b>"
```

```
'   Send ("<td><center><input type = "submit"> name = "QUERY" Value="RU Companion List"></center></td></tr><b>"
```

```
'   Send ("<tr><td BGCOLOR="#FF0000"></td><td BGCOLOR="#FF0000"></td><td BGCOLOR="#FF0000"></td></tr><b>"
```

```
'   Send ("<tr><th align = left><b><font size = +1>SE1:</th><b>"
```

```
'   Send ("<td><center><input type = "submit"> name = "QUERY" Value="Equipment - RU to RU Task"></center></td></tr><b>"
```

```
'   Send ("<tr><th align = left><b><font size = +1>SE2:</th><b>"
```

```
'   Send ("<td><center><input type = "submit"> name = "QUERY" Value="Equipment - TA Task to RU Task"></center></td></tr><b>"
```

```
'   Send ("</tr><b>"
```

```
'   Send ("<tr><th align = left><b><font size = +1>SE3:</th><b>"
```

```
'   Send ("<td><center><input type = "submit"> name = "QUERY" Value="Environmental Objects - RU to RU Task"></center></td></tr><b>"
```

```
'   Send ("</tr><b>"
```

```
'   Send ("<tr><th align = left><b><font size = +1>SE4:</th><th align=left>Environmental Objects to RU Task to TA Task</th><b>"
```

```
'   Send ("</tr><b>"
```

```
'   Send ("</table><p>")
```

```
'   Send ("<input type=submit name="Set4" value="Run Specified Query">")
```

```
'   Send ("</FORM>")
```

```
If CGI_AuthUser = "mcnett" Then
```

```
Send ("<FORM METHOD= "POST" ACTION="" /wincgi/recalc2.exe">")
```

```
Send ("<td><center><input type = "submit"> name = "RECALC" Value="RECALC"></center></td></tr><b>"
```

```
Send "</form>"
```

```
Send ("<FORM METHOD= "POST" ACTION="" /wincgi/recalcru.exe">")
```

```
Send ("<td><center><input type = "submit"> name = "RECALCRUS" Value="RECALCRUS"></center></td></tr><b>"
```

```
        Send "</form>"
    End If
    Send ("</BODY>")
    Send ("</HTML>")

    UserData.Close
    Userdb.Close
    TPSCdb.Close
    MTPTable.Close
    Db.Close
    RUdb.Close

End Sub
```

## APPENDIX L -WEIGHTING OF SME RATINGS ( $SME_F$ )

### Background

The Training Requirements Assessment Process (T-RAP) provides a means to collect multiple Subject Matter Expert (SME) ratings against Mission Training Plan (MTP) task performance measures. These ratings assess the relative degree of functionality which a simulated unit (e.g., field artillery battery) must exhibit to support the training of a WARSIM Training Audience (i.e., command and staff group). Ratings of H: High degree of detail, P: partial degree of detail, M: minimum detail, and N: not represented are applied to all performance measures under every simulated unit task as contained in the simulated unit's MTP. In cases where more than one SME rates the same task, a method was needed to account for the differences in SME expertise and experience. The concept developed, and described in this paper, is depicted in Figure 1.

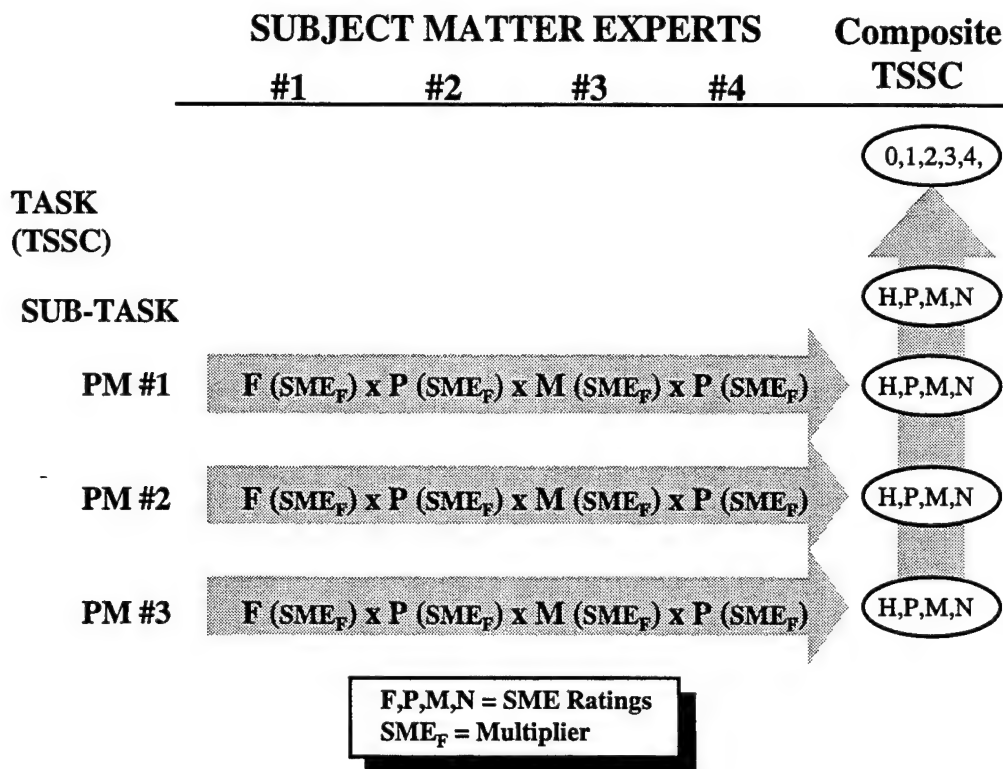


Figure L-1: Concept for Weighting SME Ratings

## Formulation

This paper describes a methodology to account for the differences between SMEs' experience in their ratings of MTP task performance measures within the T-RAP process.

The equation for calculating the SME function is:

$$(Eq. 1) \quad SME_F = F_{SA} * F_{EXP}$$

Where;

- $SME_F$  - The SME Contribution Factor ( $0 \leq SME_F \leq 1$ )
- $F_{EXP}$  - SME Experience factor ( $0 \leq F_{EXP} \leq 1$ ) and is calculated using equation 2.
- $F_{SA}$  - SME Specialty Area factor ( $0 \leq F_{SA} \leq 1$ ) is calculated using AHP methods. Within an SME's Specialty Area his ratings are unity weighted ( $F_{SA}=1$ ). Within other areas, an SME's ratings are weighted by a factor between zero and one. The Specialty Area factor accounts for the likely difference between SME ratings based on their likelihood of having knowledge of a functional area other than their own. AHP methodology was used to derive FSA for the 21 specialty areas involved in brigade training exercises. The results are displayed in Table 1.

	ADA	AG	ARM	AVN	CA	CC	CHEM	ENG	FA	FIN	INF	MED	MI	MP	ORD	PA	QM	SF	SIG	SPT	TRAN
ADA	1	.013	.142	.058	.007	.031	.013	.042	0	.026	.004	.02	.011	.005	.06	.019	.025	.057	.048	.017	.037
AG	.007	1	.028	.012	.007	.031	.003	.005	0	.231	.001	.02	.004	.003	.015	.155	.013	.014	.007	.008	.012
ARM	.063	.013	1	.104	.02	.278	.028	.047	.004	.026	.006	.02	.011	.008	.06	.019	.063	.057	.048	.034	.049
AVN	.013	.013	.142	1	.02	.056	.009	.012	.002	.026	.002	.02	.011	.008	.06	.019	.063	.057	.048	.034	.049
CA	.004	.013	.028	.012	1	.011	.019	.059	.008	.077	.014	.137	.057	.023	.3	.078	.126	.397	.144	.034	.049
CC	.021	.013	.142	.058	.04	1	.019	.015	.053	.026	.087	.02	.171	.023	.075	.019	.252	.066	.144	.203	.196
CHEM	.004	.013	.028	.012	.008	.011	1	.004	.006	.026	.012	.02	.034	.006	.075	.01	.05	.022	.024	.041	.049
ENG	.021	.013	.114	.035	.048	.089	.009	1	.035	.026	.074	.02	.069	.006	.038	.01	.05	.022	.024	.041	.049
FA	.042	.013	.114	.035	.008	.089	.009	.029	1	.026	.074	.02	.069	.006	.038	.01	.05	.022	.024	.041	.049
FIN	.005	.117	.013	.007	.002	.011	.001	.003	.004	1	.025	.02	.01	.017	.009	.039	.017	.007	.024	.008	.012
INF	.042	.013	.114	.062	.012	.100	.011	.029	.040	.026	1	.02	.069	.069	.019	.117	.034	.066	.216	.024	.049
MED	.005	.013	.013	.007	.003	.011	.001	.003	.004	.026	.025	1	.017	.023	.003	.039	.011	.008	.027	.012	.008
MI	.023	.026	.063	.035	.018	.089	.005	.01	.018	.026	.025	.02	1	.046	.003	.039	.011	.05	.108	.012	.016
MP	.023	.026	.016	.035	.018	.015	.005	.01	.018	.026	.099	.02	.034	1	.003	.039	.011	.017	.027	.012	.016
ORD	.07	.026	.016	.035	.005	.059	.021	.059	.053	.026	.033	.02	.034	.046	1	.039	.067	.033	.027	.085	.114
PA	.023	.234	.005	.012	.014	.015	.007	.012	.018	.179	.033	.02	.034	.046	.003	1	.011	.017	.027	.012	.016
QM	.07	.026	.005	.023	.041	.015	.049	.047	.053	.026	.033	.039	.034	.046	.019	.039	1	.033	.027	.085	.114
SF	.211	.026	.001	.092	.365	.015	.395	.283	.422	.026	.264	.275	.206	.138	.038	.117	.045	1	.003	.043	.029
SIG	.07	.078	.005	.023	.052	.044	.056	.047	.053	.051	.038	.039	.041	.069	.019	.039	.011	.033	1	.043	.014
SPT	.141	.156	.005	.069	.157	.015	.225	.142	.105	.051	.075	.118	.041	.138	.131	.078	.045	.011	0	1	.071
TRANS	.141	.156	.005	.277	.157	.015	.113	.142	.105	.051	.075	.118	.041	.276	.033	.078	.045	.011	0	.213	1

Table L-1.  $F_{SA}$  Factors

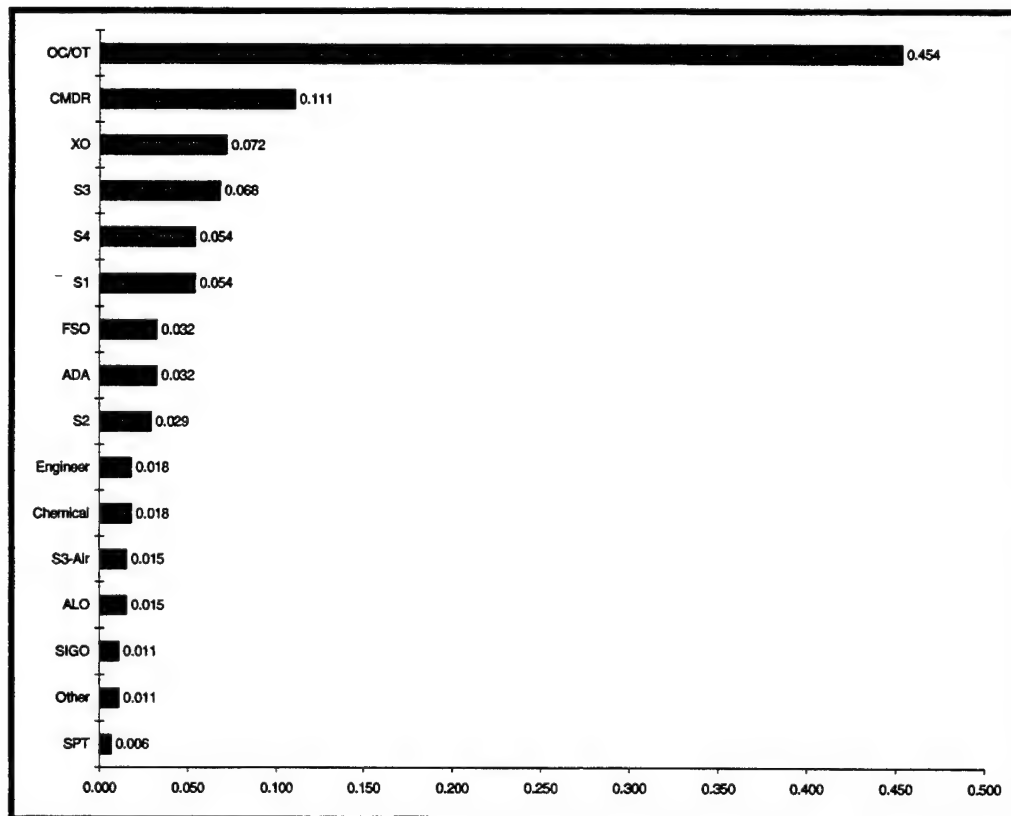
The SME experience factor ( $F_{EXP}$ ) is calculated using equation 2 thru 5 and information provided by SMEs when the log into the T-RAP system (see Figure 6).

$$(Eq. 2) \quad F_{EXP} = (V_{OC/OT} * F_{OC/OT}) + (V_{i,Trainee} * F_{i,Trainee}) + (V_{SupportStaff} * F_{SupportStaff})$$

Where;

- $V_{OC/OT}$  - Relative value of OC/OT experience compared to trainee and support staff experience.
- $F_{OC/OT}$  - Combined factor from OC/OT experience chart calculated using equation 3.
- $V_{i,Trainee}$  - Relative value of trainee experience (includes relative factors for different positions held) compared to OC/OT and support staff experience.
- $F_{i,Trainee}$  - Combined factor from Trainee experience chart calculated using equation 4.
- $V_{SupportStaff}$  - Relative value of support staff experience compared to OC/OT and trainee experience.
- $F_{SupportStaff}$  - Combined factor from Support Staff experience chart calculated using equation 5.

**NOTE:**  $V_{OC/OT}$ ,  $V_{i,Trainee}$ , and  $V_{SupportStaff}$  were determined using pairwise comparisons then normalized to 1. Their values are shown in Fig 2.



**Figure L-2. Relative Value Factors Based on Position Experience ( $V_x$ )**

For each block of OC/OT experience within the experience matrix (Figure 6),

(Eq. 2a) 
$$F_{OC/OT} = \sum X_i * Z_k$$

Where;

- $X_i$  - The  $i^{th}$  contribution factor based on the type of experience (i.e., actual deployment, CTC, BCTP, ...).
- $Z_k$  - The  $k^{th}$  contribution factor based on the echelon of experience (i.e., BDE, BN).

For each block of Trainee experience within the experience matrix,

(Eq. 2b) 
$$F_{Trainee} = \sum X_i * Z_k$$

Where;

- $X_i$  - The  $i^{th}$  contribution factor based on the type of experience (i.e., actual deployment, CTC, BCTP, ...).
- $Z_k$  - The  $k^{th}$  contribution factor based on the echelon of experience (i.e., BDE, BN).

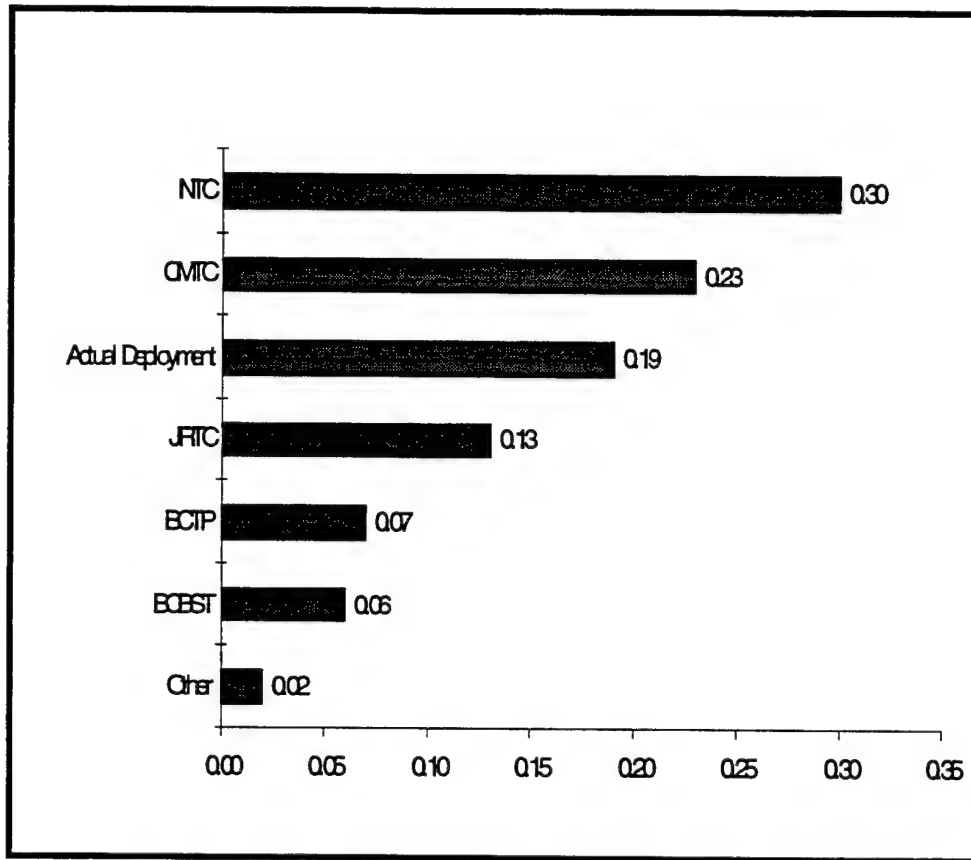
For each block of Support Staff experience within the experience matrix,

(Eq. 2c) 
$$F_{SupportStaff} = \sum X_i * Z_k$$

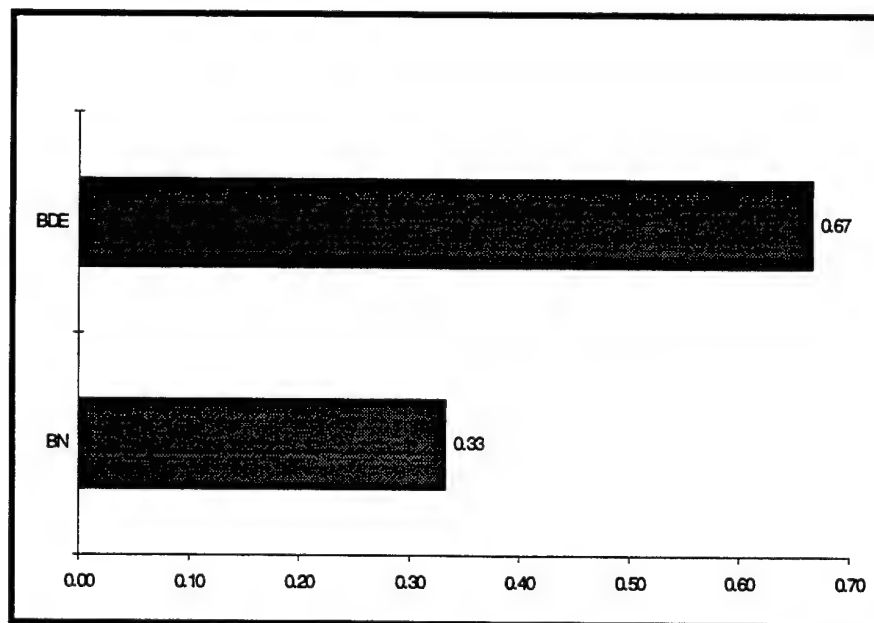
Where;

- $X_i$  - The  $i^{th}$  contribution factor based on the type of experience (i.e., actual deployment, CTC, BCTP, ...).
- $Z_k$  - The  $k^{th}$  contribution factor based on the echelon of experience (i.e., BDE, BN).

**NOTE:** The contribution factors  $x_i$ , and  $z_k$  were determined using pairwise comparison and normalized to 1. The values for these are shown on the charts in Figures 3 and 4.



**Figure L-3. Contribution Factors Based on Type of Experience (X<sub>i</sub>)**



**Figure L-4. Contribution Factors Based on Echelon of Experience (Z<sub>k</sub>)**

## Application of SME Weighting Factors

For each task step or performance measure being considered, the SME weighting factor ( $SME_F$ ) must be calculated and the individual factors normalized to determine the relative value of the contribution of each SME to that task step or performance measure rating. However, in order to maintain the established methodology for 'rolling up' SME performance measure ratings to a TPS or TSS Code at the task level, these normalized values were converted to multipliers IAW the scale in Table 2.

Normalized Range	Multiplier
0.000 to 0.049	0
0.050 to 0.199	1
0.200 to 0.399	2
0.400 to 0.599	3
0.600 to 0.799	5
0.800 to 0.899	7
0.900 to 1.000	10

Table L-2. Normalized SME Multipliers

If an SME has a normalized weighting factor of 0.631 and has rated the performance measure as moderate, then the total contribution to the roll up algorithm is 5 moderates. After each SME has had their multiplication factor assigned, the final performance measure rating is attained using the heuristic algorithm shown in Table 3 for TPSC ratings and Table 4 for the TSSC ratings.

Number of SMEs	Rating	Methodology
2 or more	High	At least 66% of the adjusted ratings are high.
	Moderate	At least 66% of the adjusted ratings are high or moderate.
	Not Required	If neither of the above constraints is met.
1		Rating of the SME is used

Table L-3. TPSC Heuristic Algorithm

Number of SMEs	Rating	Methodology
2 or more	Full	At least 66% of the adjusted ratings are Full.
	Partial	At least 66% of the adjusted ratings are Full or Partial.
	Minimal	At least 25% of the ratings are minimal.
	Not Required	If none of the above constraints is met.
1		Rating of the SME is used

Table L-4. TSSC Heuristic Algorithm

These relative factors are then used to roll up the final TPSC/TSSC code for the task step or performance measure being considered.

### Calculating one experience factor ( $F_{EXP}$ )

Using equation 2 and the values from Figure 2, Figure 3 and Table 1, the Experience Factor for each SME can be computed. As an example, consider the experience matrix in Figure 3 for one SME. To calculate the SME's  $F_{EXP}$  we first need to determine the  $V_x$  and  $F_x$  values to plug into equation 2;

$$(Eq\ 2) \quad F_{EXP} = (V_{OC/OT} * F_{OC/OT}) + (V_{i,Trainee} * F_{i,Trainee}) + (V_{SupportStaff} * F_{SupportStaff}).$$

To determine the  $F_{OC/OT}$  values we review the OC/OT portion of the matrix and find that the SME has battalion experience at the National Training Center. Using equation 3 with  $x_j = 0.30$ ,  $z_k = 0.33$  and  $V_{OC/OT} = 0.463$ , results in;

$$F_{OC/OT} = 0.3 * 0.33 = 0.100 \quad \text{and,} \quad V_{OC/OT} * F_{OC/OT} = (0.454) * (0.1) = 0.0454$$

To determine the  $F_{i,Trainee}$  values we review the Trainee portion of the matrix and find that the SME has battalion experience in actual deployment in positions of commander, S-3 and Staff ADA officer and as a commander at NTC. He also has brigade experience as an XO and Staff ADA officer in actual deployment.

	Actual Deploy.	JRTC	CMTC	NTC	BCTP	BCBST	Other
CMDR	BDE BN X	BDE BN	BDE BN	BDE BN X	BDE BN	BDE BN	BDE BN
XO	BDE BN	BDE BN	BDE BN	BDE BN	BDE BN	BDE BN	BDE BN
S1	BDE BN	BDE BN	BDE BN	BDE BN	BDE BN	BDE BN	BDE BN
S2	BDE BN	BDE BN	BDE BN	BDE BN	BDE BN	BDE BN	BDE BN
S3	BDE BN X	BDE BN	BDE BN	BDE BN	BDE BN	BDE BN	BDE BN
S3-Air	BDE BN	BDE BN	BDE BN	BDE BN	BDE BN	BDE BN	BDE BN
S-4	BDE BN	BDE BN	BDE BN	BDE BN	BDE BN	BDE BN	BDE BN
FSO	BDE BN	BDE BN	BDE BN	BDE BN	BDE BN	BDE BN	BDE BN
Staff Engineer	BDE BN	BDE BN	BDE BN	BDE BN	BDE BN	BDE BN	BDE BN
Staff ADA	BDE X BN X	BDE BN	BDE BN	BDE BN	BDE BN	BDE BN	BDE BN
Staff Chemical	BDE BN	BDE BN	BDE BN	BDE BN	BDE BN	BDE BN	BDE BN
ALO	BDE BN	BDE BN	BDE BN	BDE BN	BDE BN	BDE BN	BDE BN
SIGO	BDE BN	BDE BN	BDE BN	BDE BN	BDE BN	BDE BN	BDE BN
Other	BDE BN	BDE BN	BDE BN	BDE BN	BDE BN	BDE BN	BDE BN

	Actual Deploy.	JRTC	CMTC	NTC	BCTP	BCBST	Other
Support Staff	BDE BN	BDE BN	BDE BN	BDE BN	BDE BN	BDE BN	BDE BN
OC/OT	BDE BN	BDE BN	BDE BN	BDE BN X	BDE BN	BDE BN	BDE BN

**Figure L-3: Sample Experience Matrix**

Using equation 4 with the appropriate values for  $x_j$  and  $z_k$  from figures 1, 2 and 3, results in;

$$F_{\text{CMDR,Trainee}} = (X_{\text{Deploy}} + X_{\text{NTC}}) * Z_{\text{BN}} = (0.190 + 0.300) * 0.333 = 0.163$$

$$V_{\text{CMDR,Trainee}} * F_{\text{CMDR,Trainee}} = 0.111 * 0.163 = 0.0181$$

$$F_{\text{S-3,Trainee}} = (X_{\text{Deploy}}) * Z_{\text{BN}} = (0.19) * 0.333 = 0.0633$$

$$V_{\text{S-3,Trainee}} * F_{\text{S-3,Trainee}} = 0.068 * 0.0633 = 0.0043$$

$$F_{XO,Trainee} = (X_{Deploy}) * Z_{BDE} = (0.19) * (0.667) = 0.1267$$

$$V_{XO,Trainee} * F_{XO,Trainee} = 0.072 * 0.1267 = 0.0091$$

$$F_{ADA,Trainee} = (X_{Deploy}) * (Z_{BDE} + Z_{BN}) = (0.190) * (.667 + 0.333) = 0.19$$

$$V_{ADA,Trainee} * F_{ADA,Trainee} = 0.032 * 0.19 = 0.0061$$

$$\Sigma V_{i,Trainee} * F_{i,Trainee} = 0.0181 + 0.0043 + 0.0091 + 0.0061 = 0.0376$$

No support staff experience was reported. Summing the OC/OT and Trainee factors results in an overall  $F_{EXP}$  of;

$$F_{EXP} = (V_{OC/OT} * F_{OC/OT}) + (V_{i,Trainee} * F_{i,Trainee}) + (V_{SupportStaff} * F_{SupportStaff})$$

$$= (0.0454) + (0.0376) + (0) = 0.083.$$

This SME would be assigned an experience factor of 0.083.

## An Example

(NOTE: The factors used in this example are fictional and are not intended to match the tables and/or figures of factors presented in this paper.)

A particular task performance measure, from an Armor MTP task, has been rated by 5 SME's. Their ratings and associated Experience and Subject Matter area functions (Eq 2 and Eq 3, respectively) and raw SME weighting values (Eq 1) are shown below:

Rating	F <sub>EXP</sub>	F <sub>SA</sub>	SME <sub>EXP(r)</sub>
Moderate	0.343	0.112	0.038
Not Required	0.112	0.004	0.000
Moderate	0.091	0.215	0.020
High	0.057	1.000	0.057
Moderate	0.222	1.000	0.222

Normalized raw SME weighting factors are as shown below:

SME	SME <sub>EXP(r)</sub>	Normalized
FA SME	0.038	0.114
FIN SME	0.000	0.001
INF SME	0.020	0.058
ARM SME 1	0.057	0.169
ARM SME 2	0.222	0.658

Using the normalized values and the multiplication factors from Table 2, resulting in the following SME multipliers and adjusted performance measure ratings;

SME	Normalized	Multiplier	Rating
FA SME	0.114	1	Moderate
FIN SME	0.001	0	Not Required
INF SME	0.058	1	Moderate
ARM SME 1	0.169	1	High
ARM SME 2	0.658	5	Moderate
<b>Total</b>		1	High
		7	Moderate

The final ratings are summed as; 1 high and 7 moderate with a roll up rating of moderate.